FactoryCast User Guide
For Quantum, Premium and Micro

31001229 08 eng Version 3.1 Décembre 2005

31001229 08

a brand of Schneider Electric

Telemecanique

Table of Contents



Safety Information 9

About the Book 11

Chapter 1 Introduction to FactoryCast 13
At a Glance 13
What Is FactoryCast 14
Components of FactoryCast 15
FactoryCast Server 17
FactoryCast Configurator 18
FactoryCast Client 19
System Requirements 20
Installation 21

Chapter 2 Web Site Security 27
Web Site Security 27
Internal Security 28
External Security 30

Chapter 3 Default Web Site for Quantum 33
Default Web Site for Quantum 33
3.1 Quantum Home Page 34
Quantum Home Page 34
3.2 Monitoring Page 35
Monitoring Home Page 35
3.3 Diagnostics Page 37
Overwiew of the Diagnostics Page 37
Diagnostics Home Page 38
Configured Local Rack Page 39
CPU Configuration Page 40
Remote I/O Status Page 41
Ethernet Module Diagnostic Pages 42
NOE Properties Page 50
3.4 I/O Diagnostics 51
Overview of I/O Diagnostics 51

Configured Remote I/O Page . . . 52
Remote I/O Drop Pages . . . 54
Remote I/O Drop Module Pages . . . 56
Distributed I/O Drops Page . . . 58
Distributed I/O Specific Drop Page . . . 59
Distributed I/O Module Page . . . 60
3.5 Setup Page . . . 61
Overview of the Setup page . . . 61
Setup Home Page . . . 62
Configure SNMP Page . . . 64
Configuring Faulty Device Replacement . . . 67
Configuring the Global Data (Publish/Subscribe) Utility . . . 71

Chapter 4 Default Web Site for Premium . . . 75
Default Web Site for Premium . . . 75
4.1 Premium Home Page . . . 76
Premium Home Page . . . 76
4.2 Monitoring Page . . . 77
Monitoring Home Page . . . 77
4.3 Diagnostics Page . . . 79
Overwiew of the Diagnostics Page . . . 79
Diagnostics Home Page . . . 80
Rack Viewer Page . . . 81
PLC Personality Page . . . 82
Ethernet Module Statistics Page . . . 83
4.4 Option Modules Diagnostics . . . 91
Overview of Option Modules Diagnostics . . . 91
FIP I/O Module Diagnostics Page . . . 92
Digital I/O Module Diagnostics Page . . . 98
Analog I/O Module Diagnostics Page . . . 100
AS-i Module Diagnostics Page . . . 102
PCMCIA Communication Card Diagnostics Page . . . 104
Standard Module Diagnostics Page . . . 106
4.5 Setup Page . . . 107
Overview of the Setup page . . . 107
Setup Home Page . . . 108
Address Server Page for the HTTP Server . . . 109

Chapter 5 Common Quantum and Premium Setup Page . . . 113
Common Quantum and Premium Setup Page . . . 113
Configuring the Electronic Mail Notification Service . . . 114
Configuring the Time Synchronization Service . . . 116

Chapter 6 Default Web Site for Micro . . . 123
Default Web Site for Micro . . . 123
6.1 Micro Home Page . . . 124

Micro Home Page 124
6.2 Monitoring Page 126
Monitoring Home Page 126
6.3 Diagnostics Page 128
Overwiew of the Diagnostics Page 128
Diagnostics Home Page 129
Rack Viewer Page 131
PLC Personality Page 132
PLC Communication Pages 133
PLC Analog I/O Page 135
PLC Counter Page 136
Rack Viewer 137
Digital I/O Module Diagnostics Page 139
Analog I/O Module Diagnostics Page 140
Standard Module Diagnostics Page 141
6.4 Setup Page 142
Setup Home Page 142

Chapter 7 FactoryCast Configurator 145
FactoryCast Configurator 145
7.1 Creating a New Configuration 146
Overview of Creating a New Configuration 146
Creating a New Configuration 147
Setting Passwords 150
Setting the IP Address 153
Setting File Locations 154
Setting Default Symbol Access 155
Setting Symbol Access Level 156
Saving the Settings 157
7.2 Creating a Namespace 158
Creating a Namespace 158
Importing Symbols (Variables) 159
Setting Symbol (Variable) Access 162
Adding Direct Addresses 163
Direct Address Blocks and Symbol (Variable) Security 167
Editing and Deleting Direct Addresses 169
Synchronizing Your Namespace with Its Database 170
7.3 Transferring Files to the Web Server 172
Transferring Files to the Web Server 172
Custom Web File 173
Loading the Namespace File 176
Loading the Security File 177
Backing Up the Web Server 178
Restoring the Web Server from a Backup 179
7.4 FactoryCast Embedded Server Maintenance 180
Embedded Server Maintenance 180

Embedded Server Operations 181
Restoring Module Defaults. 184
Setting the XWay Address 186
Loading the FactoryCast Applet Plug-ins. 187

Chapter 8 Editors. 191
Editors 191
8.1 Data Editor 192
Data Editor 192
Data Editor 193
Data Editor Spreadsheet 196
Creating a Data Template 203
Saving a Data Template 204
Using an Existing Data Template. 205
Inserting a Symbol (Variable) in a Data Template 206
Inserting a Direct Address in a Data Template 207
Modifying Data Values in a Data Template 208
8.2 Data Editor Lite 209
Data Editor Lite 209
8.3 Pocket Data Editor 211
Pocket Data Editor 211
Pocket Data Editor 212
Using Pocket Data Editor 214
8.4 Graphic Editor 220
Graphic Editor 220
Graphic Editor Overview 221
Top Window User Functions 223
Display Window User Functions 228
Property Sheet 231
Security 233
Graphic Editor Applet Parameters 234
Graphic Objects 236
Extended Graphic Objects 257
8.5 Graphic Viewer 268
Graphic Viewer 268

Chapter 9 Adding Custom Pages to the Site 271
Adding Custom Pages to the Site 271
9.1 Working with Custom Pages 272
Working with Custom Pages 272
Downloading a Custom Home Page 273
Downloading Supporting Pages. 275
9.2 Using FactoryCast's Java Applets Using HTML Code. 276
Using FactoryCast's Java Applets on Custom Web Pages 276
Inserting Applets on a Web Page 277
Inserting LiveBeanApplet 278

Inserting LiveBeanApplet using HTML Code 281
Inserting LiveLabelApplet using HTML Code 284
Inserting LiveTableApplet using HTML Code 289
9.3 Using FactoryCast's Microsoft FrontPage Extension 291
Using FactoryCast's Microsoft FrontPage Extension 291
Installing FactoryCast's Microsoft FrontPage Extension 292
Inserting LiveBeanApplet Using FrontPage 298
Inserting LiveLabelApplet Using FrontPage 307
Inserting LiveTableApplet Using FrontPage 311
9.4 Using Graphic Objects Lite 315
Using Graphic Objects Lite 315
Downloading the Graphic Objects Lite Library 316
Description of Graphic Objects Lite 317

Chapter 10 Alarm Viewer 337
Alarm Viewer 337
Display 338
Operation and Management of Alarms 341
Limitations 342

Appendices 343
Appendices for FactoryCast User's Guide 343

Appendix A Browser Requirements, Settings, and Security Considerations 345
Browser Requirements, Settings, and Security Considerations 345
Browser Version 346
Browser Settings 347
Browser Security Considerations 348

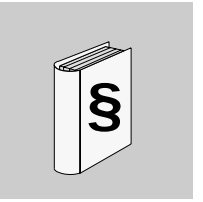
Appendix B Performance Benchmarks 349
Performance Benchmarks 349
B.1 Micro Performance Benchmarks 350
Micro Performance 350
Micro Performance Benchmarks 351
Monitoring Applet Performance 352
SOAP Performance Benchmarks 354
B.2 Premium Performance Benchmarks 355
Premium Performance 355
Premium Performance Benchmarks 356
Monitoring Applet Performance on ETY5103 359
SOAP Performance Benchmarks 361
B.3 Quantum Performance Benchmarks 362
Quantum Performance 362
Quantum Performance Benchmarks 363
Performance on NOE77111 364

SOAP Performance Benchmarks 366
B.4 FactoryCast Configurator Performance Benchmarks 367
FactoryCast Configurator Performance Benchmarks 367

Appendix C SOAP Web Services 369
Presentaton 369
Principles for Designing a SOAP Client Interface 371
ModBusXMLDA SOAP Services 372
SymbolicXMLDA SOAP Services 383
How to optimize massive symbolic requests 387
Visual Basic examples 390
Online SOAP Documentation 391

Index 393

Safety Information



Important Information

NOTICE

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.



The addition of this symbol to a Danger or Warning safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.



This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates an imminently hazardous situation, which, if not avoided, **will result** in death, serious injury, or equipment damage.

⚠ WARNING

WARNING indicates a potentially hazardous situation, which, if not avoided, **can result** in death, serious injury, or equipment damage.

⚠ CAUTION

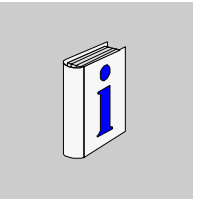
CAUTION indicates a potentially hazardous situation, which, if not avoided, **can result** in injury or equipment damage.

PLEASE NOTE

Electrical equipment should be serviced only by qualified personnel. This is not a safety product. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material. This document is not intended as an instruction manual for untrained persons.

© 2005 Schneider Electric. All Rights Reserved.

About the Book



At a Glance

Document Scope

This user guide introduces the FactoryCast software package, which is used to customize a Web site on the Embedded Server module. The site can be accessed via a browser to view and modify data from a Quantum or Premium or Micro programmable logic controller (PLC). FactoryCast provides all the Web pages and Java applets needed to view run-time data from a controller. FactoryCast offers two levels to customize the default Web site.

- Create a Web-enabled database to view and modify the run-time values of symbols (variables) and direct addresses
- Add Web pages to the site

FactoryCast provides all the tools needed to maintain a Web site on the Embedded Server, including methods for downloading, backing up, and restoring files.

Validity Note

The data and illustrations found in this book are not binding. We reserve the right to modify our products in line with our policy of continuous product development. The information in this document is subject to change without notice and should not be construed as a commitment by Schneider Electric.

Related Documents

Title of Documentation	Reference Number
Quantum NOE 771 xx Ethernet Modules User Guide	840USE11600
PL7 Junior/Pro Setup Diagnostic Functions	TLXDSDIAGPL7
TSX Micro TSX ETZ 410/510 Modules User Manual	ETZ410_510_E (on FactoryCast CD-ROM)

Product Related Warnings

Schneider Electric assumes no responsibility for any errors that may appear in this document. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

No part of this document may be reproduced in any form or by any means, electronic or mechanical, including photocopying, without the express written permission of Schneider Electric.

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When controllers are used for applications with technical safety requirements, please follow the relevant instructions.

Failure to use Schneider Electric software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this product related warning can result in injury or equipment damage.

User Comments

We welcome your comments about this document. You can reach us by e-mail at techpub@schneider-electric.com

Introduction to FactoryCast

1

At a Glance

Purpose

This chapter describes FactoryCast for Quantum and Premium and Micro, including its functions, components and system requirements.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
What Is FactoryCast	14
Components of FactoryCast	15
FactoryCast Server	17
FactoryCast Configurator	18
FactoryCast Client	19
System Requirements	20
Installation	21

What Is FactoryCast

Overview	FactoryCast is a software package that you use to customize a Web site on the Embedded Web Server module. The site can be accessed via a browser to view and modify data from a Quantum or Premium or Micro programmable logic controller (PLC).
FactoryCast Web Site	FactoryCast provides all the Web pages and Java applets you need to view run-time data from your controller. You can use the FactoryCast default Web site simply by configuring the module and accessing it with a browser over the intranet.
Custom Web Site	FactoryCast gives you two levels to customize the default Web site embedded in the PLC module: ● create a Web-enabled database to view and modify the run-time values of PLC symbols (variables) and direct addresses, ● add your own Web pages to the site.
Site Maintenance	FactoryCast also provides all the tools you need to maintain your Web site on the Embedded Server, including ways to download, backup, and restore files.

Components of FactoryCast

Overview

The three components of FactoryCast are:

- FactoryCast Server,
- FactoryCast Configurator,
- FactoryCast Client.

Quantum Version

For Quantum, the FactoryCast components are integrated as follows.

This Component...	Is Used To...
140 NOE 211 10 140 NOE 251 10 (Fiber Optic)	Store the Web site on its Embedded Server.
140 NOE 771 1X	Enhance Web server performance and memory.
Concept v. 2.1 or higher Modsoft v. 2.5 and higher Unity Pro v 1.0 and higher	Program the controller.
FactoryCast Configurator	Configure the Web site, download data and maintain the site on the server.
FactoryCast Client	Display and modify PLC data.
Netscape Communicator 4.06 or higher Internet Explorer 4.0 (w/Service Pack 2) or higher	View the Web site.

Premium Version

For Premium, the FactoryCast components are integrated as follows.

This Component...	Is Used To...
TSX ETY 110 WS	Store the Web site on its Embedded Server.
TSX ETY 510	Enhance Web server performance and memory
PL7 v. 3.0 and higher Unity Pro v1.0 and higher	Program the controller.
FactoryCast Configurator	Configure the Web site, download data and maintain the site on the server.
FactoryCast Client	Display and modify PLC data.
Netscape Communicator 4.06 or higher Internet Explorer 4.0 (w/Service Pack 2, minimum) or higher	View the Web site.

Micro Version

For Micro, the FactoryCast components are integrated as follows.

This Component...	Is Used To...
TSX ETZ 510	Store the Web site on its Embedded Server and enhance Web server performance and memory.
PL7 Software	Program the controller. No specific requirement on PL7 software relative to FactoryCast.
FactoryCast Configurator	Configure the Web site, download data and maintain the site on the server.
FactoryCast Client	Display and modify PLC data.
Netscape Communicator 4.06 or higher Internet Explorer 4.0 (w/Service Pack 2, minimum) or higher	View the Web site.

FactoryCast Server

Overview

FactoryCast server consists of HTTP and FTP servers embedded in a Quantum or Premium or Micro Ethernet option module.

How it Works

The servers contain a default set of diagnostic Web pages and Java applets. The user can add custom Web pages and applets for specific applications.

Versions

FactoryCast server is embedded on the following PLC modules.

Part Number	Description	Memory Available for Customization
140 NOE 211 10	Quantum Embedded Server Module with Ethernet TCP/IP, 10 BaseT Twisted Pair, 1 Channel, and FactoryCast Configurator Software	Configurable
140 NOE 251 10	Quantum Embedded Server Module with Ethernet TCP/IP, 10 BaseFL Fiber Optic, 1 Channel, and FactoryCast Configurator Software	Configurable
140 NOE 771 1X	Quantum FactoryCast module with Ethernet TCP/IP 10/100 BaseT twisted pair/100 Base Fx fiber optic, 1 channel, and FactoryCast configurator software	up to 8 Mb
TSX ETY 110 WS	Premium Embedded Server Module with Ethernet TCP/IP, 10 BaseT Twisted Pair, AUI and FactoryCast Configurator Software	Configurable
TSX ETY 510	Premium Web Server Module with Fast Ethernet 10/100 BaseT Twisted Pair, and FactoryCast Configurator Software	up to 8 Mb
TSX ETZ 510	Micro Web server device accessible via Fast Ethernet 10/100 BaseT Twisted Pair or remotly via PPP server/Modem	up to 8 Mb

FactoryCast Configurator

Overview

You can use the FactoryCast Configurator to configure and maintain your Web site.

You also can use FactoryCast to create a Web-enabled database of variables (symbols) and direct addresses, which can be viewed and modified during run-time over the Web.

Another function of the Configurator is to provide general FTP capabilities enabling you to load Custom Web pages to the FactoryCast module.

Configuring a Site

The configuration tool helps you to:

- set security, including passwords and read/write protection,
- add your own Web pages, images and Java applets to a site,
- download and upload files to the Embedded Server.

Creating a Database

The configuration tool allows you to create a Web-enabled database using symbols (variables) and direct addresses from your Concept, PL7 or Unity Pro database. You can use this Web-enabled database to view and modify the value of symbols (variables) and direct addresses while the controller is running.

Maintaining a Site

The configuration tool allows you to back up files, restore files and, if necessary, reflash FactoryCast Configurator files to the Embedded Server.

FactoryCast Client

Overview

The FactoryCast client offers run-time diagnostics via predefined Web pages. The features available are:

- a Data Editor,
- a Graphics Editor,
- an Alarm Viewer,
- a Rack Viewer.

Rack Viewer

The Rack Viewer allows you to display the status and configuration of the controller, embedded server module, other option modules and I/O modules.

Data Editor

The Data Editor allows you to view and modify variables (symbols and direct addresses).

Graphic Editor

The Graphic Editor allows you to create and view graphical objects. Each graphical object can be linked to a variable or address in the embedded server.

Alarm Viewer

When the PLC application has diagnostic properties activated, the Alarm Viewer allows you to display application faults.

System Requirements

Overview

This section provides minimum system requirements for FactoryCast Configurator software. If FactoryCast Configurator programs are used simultaneously with other software packages, a more powerful configuration may be required.

To Use the Configuration Tool

These are the minimum system requirements.

Processor	Pentium 166 Mhz (Pentium 200 Mhz recommended)
Operating System	Windows NT 4.0 (w/Service Pack 3), Windows 2000, and Windows XP
Ram	32 Mb (64 MB recommended)
Drives	Hard Disk (free space) 40 Mb 4XCD-ROM
Monitor	SVGA 800x600

To View and Modify Run-time Diagnostics

These are the minimum system requirements.

Processor	Pentium 166 Mhz (Pentium 200 Mhz recommended)
Operating System	Windows NT 4.0 (w/Service Pack 3), Windows 2000, and Windows XP
Ram	32 Mb (64 MB recommended)
Monitor	SVGA 800x600
Browsers	Netscape Communicator 4.06 or higher or Internet Explorer 4.0 (w/Service Pack 2) or higher

Browsers

For browser requirements, see *Browser Version, p. 346*.

Installation

Overview

FactoryCast comes on a CD and runs automatically. Once the CD has loaded on your PC, follow the installation dialogs.

Quantum Downloaded Files before V3.1

FactoryCast downloads the following files for Quantum modules during installation.

Rack Viewer	Data Editor	Graphic Editor	Real-Time Communication
wwwroot/classes » Sys Diag.jar wwwroot/conf » Gcnftcop.sys wwwroot/images » module.gif » miniplc.gif » eight_io.gif » empty.gif » hiendcpu.gif secure/system » ctrlstat.htm » ethernet.htm » plccfg.htm » riostat.htm	wwwroot/classes » RDE.jar secure/system » rde.htm	wwwroot/classes » GDE.jar » Widgets.jar secure/system » gde.htm wwwroot/images » key.gif	wwwroot/classes » SAComm.jar

Quantum Downloaded Files in V3.1 (only for NOE77111 V3.1)

FactoryCast downloads the following files for Quantum modules during installation.

Rack Viewer	Data Editor	Graphic Editor	Real-Time Communication
wwwroot/classes » SysDiag.jar wwwroot/conf » Gcnftcop.sys wwwroot/images » module.gif » miniplc.gif »eight_io.gif » empty.gif »hiendcpu.gif wwwroot/secure/system » ctrlstat.htm » ethernet.htm » plccfg.htm » riostat.htm	wwwroot/classes » RDE.jar secure/system » rde.htm	wwwroot/classes » GDE.jar » Widgets.jar » user.jar wwwroot/secure/system » gde.htm wwwroot/images » key.gif	wwwroot/classes » SAComm.jar

Graphic Viewer	Data Editor Lite	Alarm Viewer	Pocket Data Editor
wwwroot/classes » GDV.jar » Widgets.jar » user.jarw wwroot/secure/system » gdv.htm wwwroot/images » key.gif	wwwroot/classes » rdelite.jar wwwroot/secure/system » rdelite.htm	wwwroot/classes » RAE.jar wwwroot/secure/system » rae.htm	wwwroot/cabs » RdePocket.cab wwwroot/secure/system » rdepocket.htm

Premium Downloaded Files before V3.1

FactoryCast downloads the following files for Premium modules during installation.

Rack Viewer	Data Editor	Graphic Editor	Alarm Viewer	Real-Time Communication
wwwroot/classes » JL.jar » SysDiag.jar wwwroot/conf » business.pbf » modules.pbf » products.pbf » ranges.pbf wwwroot/images » anim1.gif » backbtn.gif secure/system » plccfg.htm » ethernet.htm	wwwroot/classes » RDE.jar secure/system » rde.htm	wwwroot/classes » GDE.jar » Widgets.jar secure/system » gde.htm wwwroot/images » key.gif	wwwroot/classes » RAE.jar secure/system » rae.htm wwwroot/images » ack.gif » ackall.gif » dfb.gif » gr7.gif » help.gif » nack.gif » stop.gif » sys.gif » trasall.gif » trash.gif » asi.gif	wwwroot/classes » SAComm.jar

Premium Downloaded Files in V3.1 (only for ETY510 V3.1)

FactoryCast downloads the following files for Premium modules during installation.

Rack Viewer	Data Editor	Graphic Editor	Alarm Viewer	Real-Time Communication
wwwroot/classes » JL.jar » SysDiag.jar wwwroot/conf » business.pbf » modules.pbf » products.pbf » ranges.pbf wwwroot/images » anim1.gif » backbtn.gif secure/system » plccfg.htm » ethernet.htm	wwwroot/classes » RDE.jar secure/system » rde.htm	wwwroot/classes » GDE.jar » Widgets.jar » user.jar wwwroot/secure/system » gde.htm wwwroot/images » key.gif	wwwroot/classes » RAE.jar wwwroot/secure/system » rae.htm	wwwroot/classes » SAComm.jar

Graphic Viewer	Data Editor Lite	Pocket Data Editor
wwwroot/classes » GDV.jar » Widgets.jar » user.jar wwwroot/secure/system » gdv.htm wwwroot/images »key.gif	wwwroot/classes » rdelite.jar wwwroot/secure/system » rdelite.htm	wwwroot/cabs » RdePocket.cab wwwroot/secure/system » rdepocket.htm

Micro Downloaded Files before V3.1

FactoryCast downloads the following files for Micro modules during installation.

Rack Viewer	Data Editor	Graphic Editor	Real-Time Communication
Wwwroot/classes » JL.jar » SysDiag.jar wwwroot/conf » business.pbf » modules.pbf » products.pbf » ranges.pbf wwwroot/images » anim1.gif » backbtn.gif wwwroot/secure/system » plccfg.htm » ethernet.htm	wwwroot/classes » RDE.jar wwwroot/secure/system » rde.htm	wwwroot/classes » GDE.jar » Widgets.jar wwwroot/secure/system » gde.htm wwwroot/images » key.gif	wwwroot/classes » SAComm.jar

Micro Downloaded Files in V3.1 (only for ETZ510 V3.1)

FactoryCast downloads the following files for Micro modules during installation.

Rack Viewer	Data Editor	Graphic Editor	Real-Time Communication
Wwwroot/classes » JL.jar » SysDiag.jar wwwroot/conf » business.pbf » modules.pbf » products.pbf » ranges.pbf wwwroot/images » anim1.gif » backbtn.gif wwwroot/secure/system » plccfg.htm » ethernet.htm	wwwroot/classes » RDE.jar wwwroot/secure/system » rde.htm	wwwroot/classes » GDE.jar » Widgets.jar » user.jar wwwroot/secure/system » gde.htm wwwroot/images » key.gif	wwwroot/classes » SAComm.jar

Graphic Viewer	Data Editor Lite	Pocket Data Editor
wwwroot/classes » GDV.jar » Widgets.jar » user.jar wwwroot/secure/system » gdv.htm wwwroot/images » key.gif	wwwroot/classes » rdelite.jar wwwroot/secure/system » rdelite.htm	wwwroot/cabs » RdePocket.cab wwwroot/secure/system » rdepocket.htm

Web Site Security

2

Web Site Security

Overview

Before you set up your Web site, you should give some thought to security. While data in a default Web site is read-only, data in a custom site may be write-enabled. You should consider carefully who has access to the site and which data can be modified

This chapter discusses security concerns and some security mechanisms available to Web Utility users.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Internal Security	28
External Security	30

Internal Security

Overview

The Web site you create with FactoryCast may be accessed over an intranet. FactoryCast provides two mechanisms to ensure that only authorized users view and modify your data.

Security Mechanisms

On intranets, FactoryCast Configurator provides security through:

- Password entry
- Write restrictions



CAUTION

UNAUTHORIZED SECURITY ACCESS

Keep strict control of access to the FactoryCast Configurator software. Anyone who has access to a Configuration Tool and to your Embedded Server can override your security settings and download new settings to the server. Unauthorized or incorrect changes to data may change the behavior of your application in ways that may be undesirable or even hazardous.

Failure to follow this instruction can result in injury or equipment damage.

Password Entry

Although you may add unprotected Web pages to the site, the default Web pages and any other pages you choose to protect can only be viewed by users who supply the correct user name and password.

Write Restrictions

In order to modify your Web site with the Data Editor or Graphic Editor, a user must enter another password for write access.

In addition, users who enter the write password can only modify variables (symbols) and direct addresses which are write-enabled. When you create a Web-enabled database of variables and direct addresses, you can designate each element as read-only or write-enabled.



CAUTION

UNAUTHORIZED CHANGES TO VARIABLES OR DIRECT ADDRESSES

Be careful about which variables (symbols) and direct addresses you allow to be modified online, and be cautious about who has permission to modify them. Unauthorized or incorrect changes to data may change the behavior of your application in ways that may be undesirable or even hazardous.

Failure to follow this instruction can result in injury or equipment damage.

Security Overrides

Because the passwords and read/write settings are downloaded to the Embedded Server with the FactoryCast Configurator, anyone who has a copy of the Configurator software and access to your Embedded Server over the network can modify your security settings by downloading new ones.

External Security

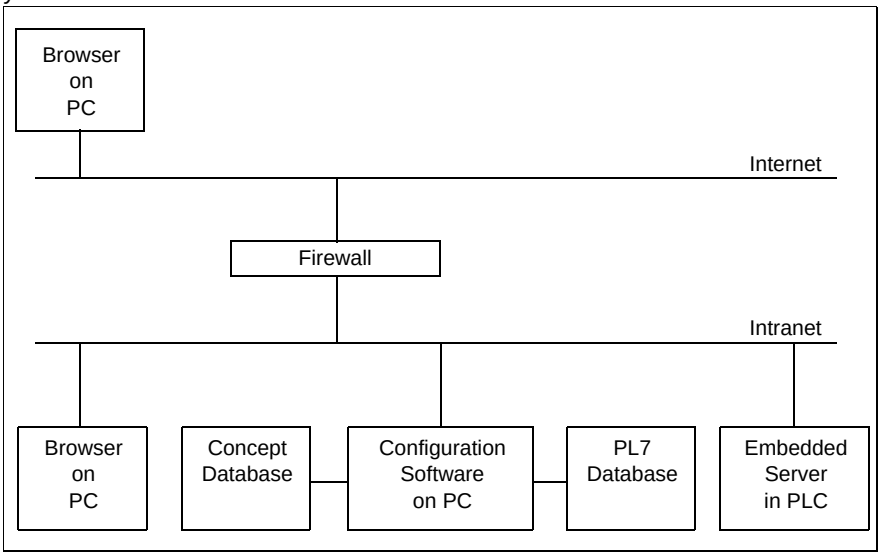
Overview

If your network is configured to allow users to view your site over the Internet, you have the same security concerns as for an intranet, but you have an extra mechanism to deal with them: a firewall.

Firewall Architecture

A firewall is a gateway from the Internet to your Embedded Server as depicted below. You can use a firewall to restrict or deny access to your Web site.

This diagram demonstrates how a firewall interacts with the Embedded Server and your PC.



Types of Firewalls

There are two types of firewalls:

- network-level firewalls,
- application-level firewalls.

Network-Level Firewalls

Network-level firewalls are frequently installed between the Internet and a single point of entry to an internal, protected intranet or network.

Application-Level Firewalls

An application-level firewall acts on behalf of an application; for instance, FTP. It intercepts all traffic destined for that application and decides whether to forward that traffic to the application. Application-level firewalls reside on individual host computers.

Considerations for FactoryCast

FactoryCast Configurator uses FTP to access Embedded Server files. If you want viewers to be able to access your site from the Internet and your Embedded Server is protected by a firewall, then that firewall must be configured to allow FTP traffic.

The firewall may be configured to allow network connections to a restricted port range or to allow traffic to and from certain IP addresses. Firewalls configured to allow incoming data to FTP's well-known TCP/IP port of 21, and to allow incoming data to ports higher than 1024, will grant access to protected Embedded Servers.

The FactoryCast client follows the "Firewall Friendly FTP" standard, RFC 1579. It issues an FTP PASV command to the FactoryCast server before all attempts to establish an FTP data connection.

FactoryCast uses TCP/IP port 80 for HTTP access to Web pages stored on an Embedded Server. Schneider Electric's MBAP protocol is used to access run-time data on TCP/IP port 502. These ports must also be made available through the firewall.

Note: Quantum NOE 211 10 users who want to add a single FTP password to the server can create an ASCII file, ftplist.dat. This file should contain in the user name string on the first line and a password string on the second line.
For example:

- MyUser,
- MyPassword.

Save this file to your local PC directory under \FactoryCast\Software\noe211\wwwroot\ftplist.dat. Next, use the FactoryCast Configurator to "Restore Defaults."
Users of Quantum NOE 771, Premium ETY 510 and Micro ETZ 510 can set an FTP password by using the Web page at: http://hostname/secure/embedded/ftp_passwd_config.htm.

Default Web Site for Quantum

3

Default Web Site for Quantum

Overview

When you receive the Embedded Server, it already contains a default Web site with diagnostic pages, Data Editor, and Graphic Editor.

You may view these pages and view direct addresses in the editors simply by installing the module and configuring its IP address. To access the site, type the IP address of the module in your browser and enter the default user name and password of "USER". However, Schneider Electric recommends that you complete the setup procedures as outlined in *FactoryCast Configurator, p. 145*.

This section describes the pages in the default Web site.

What's in this Chapter?

This chapter contains the following sections:

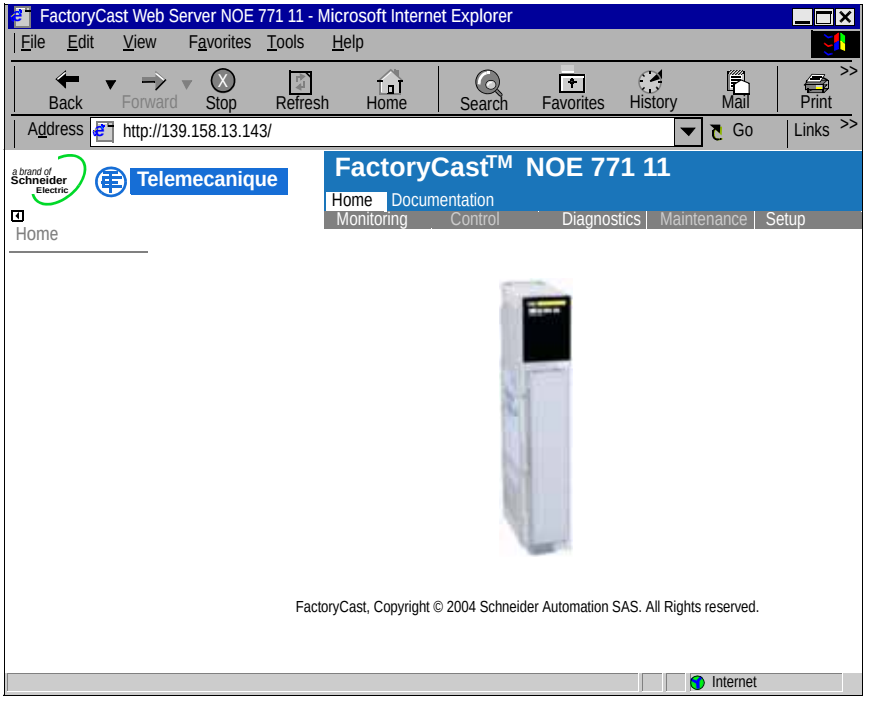
Section	Topic	Page
3.1	Quantum Home Page	34
3.2	Monitoring Page	35
3.3	Diagnostics Page	37
3.4	I/O Diagnostics	51
3.5	Setup Page	61

3.1 Quantum Home Page

Quantum Home Page

Home Page

The visitor accesses the Quantum home page by entering the IP address of the module in his web browser. No password is required to display the home page:



Links

From the Quantum home page, you can access the following pages :

- Monitoring (See *Monitoring Home Page, p. 35*),
- Diagnostics (See *Diagnostics Home Page, p. 38*),
- Setup (See *Setup Home Page, p. 62*),
- Documentation.

The visitor will have to supply a user name and a password to access the services on these pages.

3.2 Monitoring Page

Monitoring Home Page

Home Page	This page lists the various viewer services supported by the default Web site of the module and provides links for accessing the services you require.

Illustration

The Monitoring home page looks like this:



To access a service, click on a link. The services include:

- Data editor (See *Data Editor, p. 192*): for creating variable data tables to determine their values when the table is animated.
- Data editor lite (See *Data Editor Lite, p. 209*): for creating variable data tables to determine their values when the table is animated. (This editor contains fewer features than the standard Data editor.)
- Graphic editor (See *Graphic Editor, p. 220*): for creating graphics to determine the values of variables when the graphic is animated.
- Graphic viewer (See *Graphic Viewer, p. 268*): for viewing graphics to determine the values of variables when the graphic is animated.
- Password-protected custom pages (See *Adding Custom Pages to the Site, p. 271*): for viewing protected screen pages created by the user.
- Custom pages without password protection (See *Adding Custom Pages to the Site, p. 271*): for viewing unprotected screen pages created by the user.

3.3 Diagnostics Page

Overwiew of the Diagnostics Page

Overview

This section describes the different services proposed by the ‘Diagnostics’ page.

What's in this Section?

This section contains the following topics:

Topic	Page
Diagnostics Home Page	38
Configured Local Rack Page	39
CPU Configuration Page	40
Remote I/O Status Page	41
Ethernet Module Diagnostic Pages	42
NOE Properties Page	50

Diagnostics Home Page

Home Page

This page lists the various services supported by the default Web site of the module and provides links for accessing the services you require.

Illustration

The **Diagnostics** home page looks like this:



Links

To access the service you require, click on a link:

- Configured Local Rack,
- Controller Status (See *Sample Page, p. 40*),
- RIO Status (See *Remote I/O Status Page, p. 41*),
- Alarm Viewer (See *Alarm Viewer, p. 337*),
- NOE Diagnostics (See *Ethernet Module Diagnostic Pages, p. 42*),
- NOE Properties (See *NOE Properties Page, p. 50*).

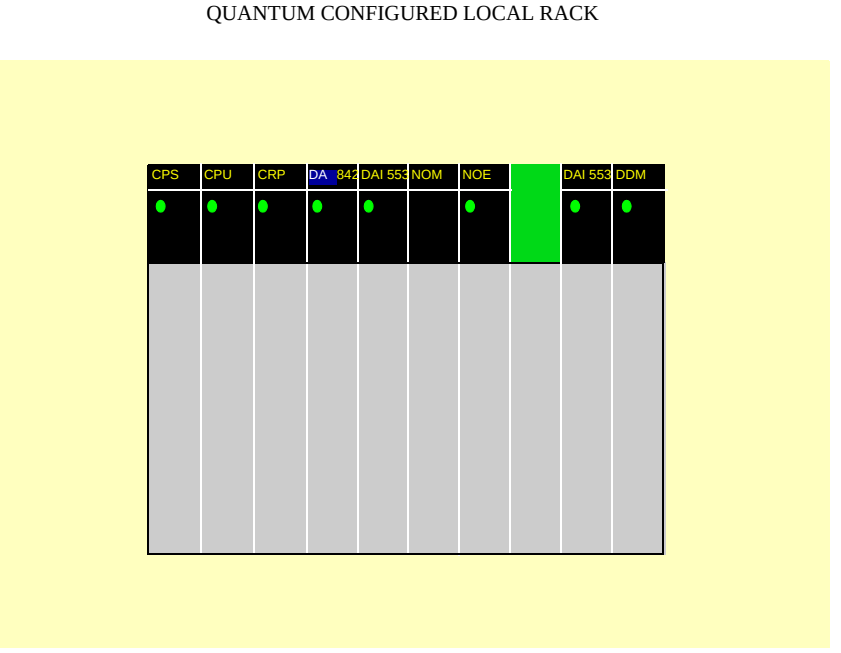
Configured Local Rack Page

Overview

The Quantum Configured Local Rack page displays the current configuration of the local rack, including the controller, Embedded Server module and any I/O modules. The rack can contain up to 16 slots.

Sample Page

Here is an example of a Configured Local Rack page.



Data

Each module is displayed in its configured slot in the rack and the following information is provided.

- A label at the top of the module tells what type it is. Question marks indicate that the module type is unknown or the slot is empty.
- An LED below the label reports the module status:
  - Green indicates that the module is functioning properly,
  - Red indicates that the module is not functioning properly.

Links

If you click on any of the modules, you will reach another Web page with detailed information about that module.

CPU Configuration Page

Overview

The **CPU Configuration Page** provides up-to-date information about the controller and its configuration. Access this page by selecting the CPU model from the Configured Local Rack or the hyperlink Controller Status (See *Sample Page, p. 40*).

Sample Page

Here is an example of a CPU Configuration page.

QUANTUM CONTROLLER STATUS

Status:	Running	Reference:	CPU 534 14
Battery:	OK	Product Type:	Quantum
Rack:	1	Exec ID:	883
Slot:	2	Logged In:	No

Description		Registers		ASCII	
System Memory [Kb]	64 Kb	0xxxxx	000001-001536	Total Words	0
Extended Memory [Kb]	96 Kb	1xxxxx	100001-100512	Total Messages	0
Total Memory [Bytes]	163840	3xxxxx	300001-300512	Word Used	0
I/O Map Words	161	4xxxxx	400001-401872	Messages Used	0
Segments	32	6xxxxx	600000-	Available Words	0
DCP Drop ID	0	Battery Coil	0-----	Available Mes-	0
Memory Protect	Off	Timer Register	4-----	# ASCII Ports	0
Constant Sweep	Off	Time of Day	4-----	ASCII Inputs	4------
Optimize	No	Stopped Codes	0x0000	ASCII Outputs	4------

Dynamic Data

Some of the data provided on this page is dynamic. Dynamic data is constantly refreshed at a rate determined by the performance of the Embedded Server, network, and client CPU.

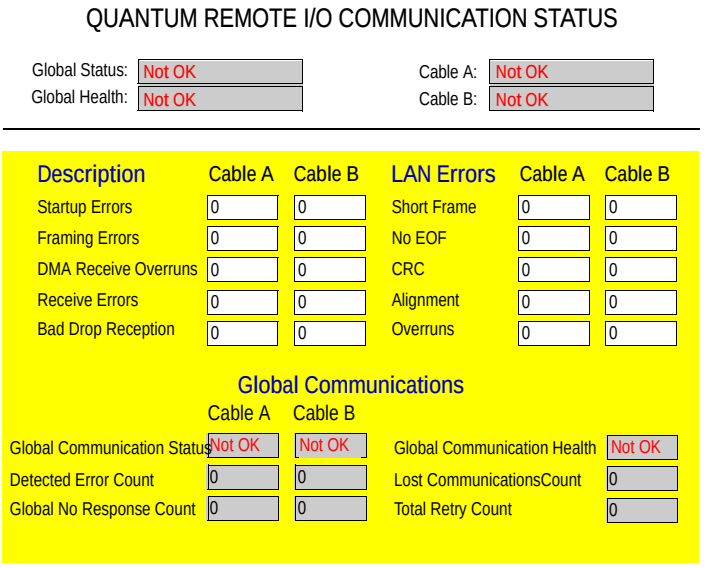
Remote I/O Status Page

Overview

The Remote I/O Status page gives an overview of the status and health of the Remote I/O network communications. Access this page by selecting the CRP Drop down menu item, "Remote I/O Status," or use the hyperlink ‘RIO Status’.

Sample Page

Here is an example of a Remote I/O Status page.



Dynamic Data

Some of the data provided on this page is dynamic. Dynamic data is constantly refreshed at a rate determined by the performance of the Embedded Server, network, and client CPU.

Ethernet Module Diagnostic Pages

Overview

The **NOE Diagnostics** menu contains a list of links for accessing the different diagnostic pages for the Ethernet module:

- Global Data utility,
- I/O scanning utility,
- Messaging utility,
- Bandwidth monitoring utility,
- Ethernet module statics,
- NTP utility,
- Email utility.

A link also allows the uploading of the private MIB source file.

Click on a link to access the desired diagnostics page.

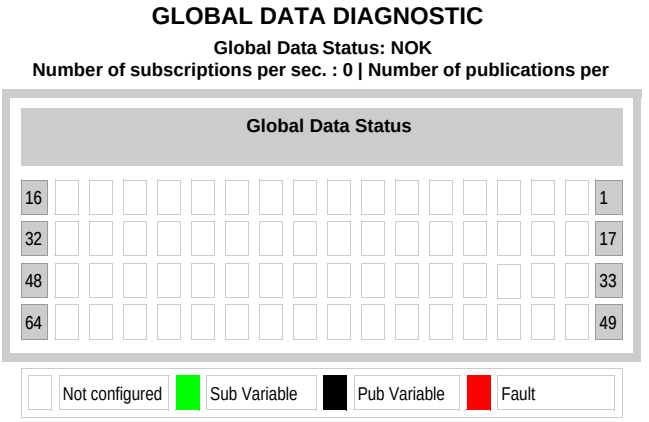
Global Data Page

Information on the general diagnostics of Global Data can be found at the top of this page:

- Status
- Number of publications per second
- Number of subscriptions per second

This page also shows a table of all published and subscribed variables in the same distribution group. Each variable is identified by its Identifier:

- Green for the subscribed variables
- Black for the published variables
- White for all unconfigured variables
- Red for variables with communication faults



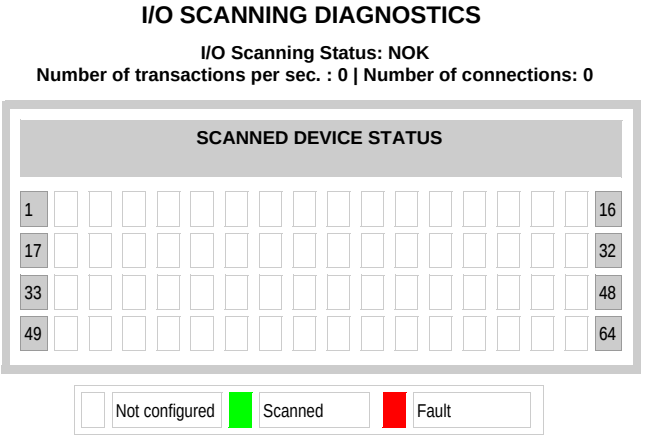
I/O Scanning Page

Information on the general diagnostics of the I/O scanning utility can be found at the top of this page:

- Status,
- Number of transactions per second,
- Number of connections.

This page also displays a summary of the status of all modules:

- Green for the **scanned** modules,
- White for the **unconfigured** modules,
- Red for **faulty** modules.



Note:
The data status (OK/NOK) indicates only the state of your local configured network:

- OK: The table reports the state of scanned (remote) devices,
- NOK: Your local system is "not OK," so data in the table is incosequential.

Messaging Page

This page provides current information on the open TCP connections on port 502.

The number of sent/received messages on the port can be found at the top of this page.

A table provides, for each connection (numbered from 1 to 64):

- the remote IP Address,
- the remote TCP port,
- the local TCP port,
- the number of messages sent from this connection,
- the number of messages received from this connection,
- the error number on this connection.

MESSAGING DIAGNOSTICS

Number of Messages sent: 2007 | Number of Messages received: 2007

Conn.	Remote address	Remote port	Local Port	Mess. sent	Mess. received	Error sent.
1	192.168.2.10	1240	502	356	356	0
2	139.168.2.10	1247	502	56	56	0

Note: Following a request to close a connection, the PLC may hold the connection open in its memory for a few minutes, during which the table will reflect the open connection.
Number of Messages received is not reset after a port 502 connection is closed. Therefore, the count indicates the total number of messages that have been received since the module was started.

Bandwidth Monitoring Page

This page displays the load distribution of the Embedded Server module between the Global Data utilities, I/O Scanning, Messaging, and other utilities:

BANDWIDTH MONITOR



Global data: 0 | I/O Scanner: 0 | Messaging: 2 | Others: 98

Global Data I/O Scanning Messaging Other

Ethernet Statistics Page

The Ethernet Module Statistics page provides information about the status, transmit and receive statistics, and errors for the Embedded Server module. Access this page by selecting the NOE module form the local rack or use the hyperlink Statistics. The following graphic is an example **Ethernet Module Statistics** page:

ETHERNET MODULE STATISTICS

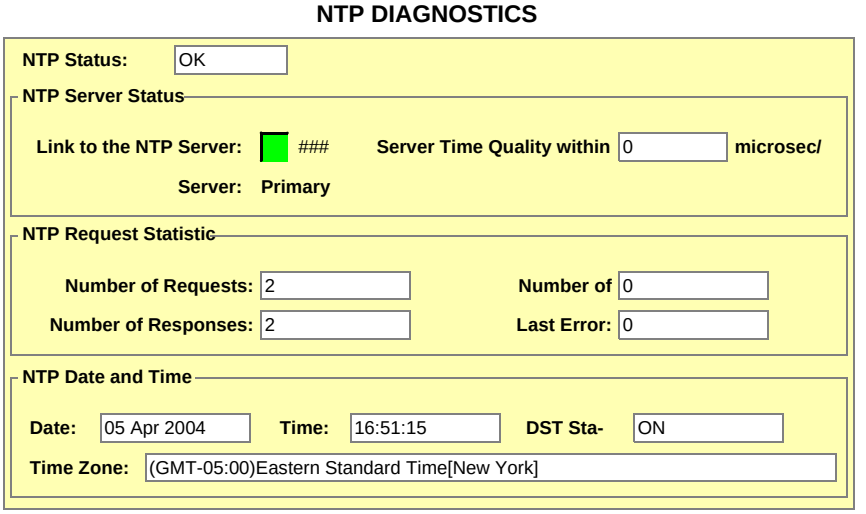
Status:	Running Link Appl	Host Name:	139.158.13.143
Reference:	140 NOE 771 10	MAC Address:	00 00 54 10 20 ae
Rack:	1	IP Address:	139.158.13.143
Slot:	Unknown	Subnet Mask:	Unknown
Transmit Speed:	10 MB	Gateway Address:	Unknown

Transmit Statistics		Receive Statistics		Functioning Errors	
Transmits	13161058	Receives	24446416	Missed Packets	0
Transmit Retries	0	Framing Errors	0	Collision Errors	0
Lost Carrier	1	Overflow Errors	0	Transmit Timeouts	0
Late Collision	0	CRC Errors	0	Memory Errors	0
Transmit Buffer Errors	0	Receive Buffer Errors	0	Net Interface Restarts	0
Silo Underflow	0				

Reset

Note: Refer to *Modicon Quantum Ethernet TCP/IP Module User Guide* 840 USE 107 00 and *Modicon Quantum Ethernet TCP/IP Module User Guide* 840 USE 115 00 for definitions of terms.

NTP Diagnostics Page



Time synchronization service parameters:

Parameter	Description
NTP status	Service is correctly configured (OK)
NTP server status	NTP client is connected to the NTP server, and if the server is Primary or Standby
NTP requests	Total number of client requests sent to the NTP server
NTP responses	Total number of server responses sent from the NTP server
Number of errors	Total number of unanswered NTP requests
Last error code	Last error code received from the NTP client
Date	Date in d/m/y format
Time	Time
Time zone	Time zone plus or minus Universal Time, Coordinated (UTC)
DST	Daylight saving time (DST) parameter is either **1.** on (enabled) **2.** off (disabled)

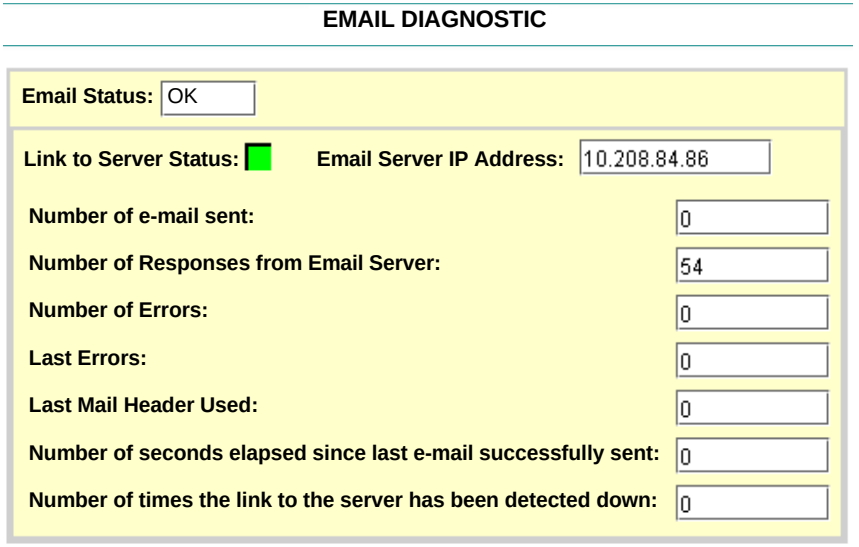
Last Error field displays values, which indicate the type of error.

Type of Error	Value
Component OK and running	0
Excessive network traffic or server overload	1
Bad parameters in the configuration	3

Type of Error	Value
Component is disabled	4
Incorrect IP	9
Time zone file absent	14
Syntax error in the customrules file	15

Email Diagnostics Page

The dialog:



Electronic mail notification service parameters

Parameter	Description
Email status	Email service is correctly configured (OK).
Link to Server Status	Ethernet module is connected to the SMTP server. Status is checked at start-up and at least every 30 minutes after start-up: ● Green = module connected to server, ● Red = module NOT connected to server.
SMTP Server IP Address	IP address of the SMTP server.
Number of e-mails sent	Total number of emails sent successfully.
Number of Responses from SMTP Server	Total number of SMTP messages received from the SMTP server.
Number of Errors	Total number of e-mails NOT sent because of an error.
Last Errors	Reason for the last error with a code in hexadecimal. 0 displays if no error occurs.
Last Mail Header Used	Last header used by giving the number.
Number of seconds elapsed since last e-mail successfully sent	Counts the number of seconds since the last email was successfully sent.
Number of times the link to the server has been detected down	Number of times the SMTP server could not be reached. (Link checked every 30 minutes.)

Crash Log Diagnostics Page

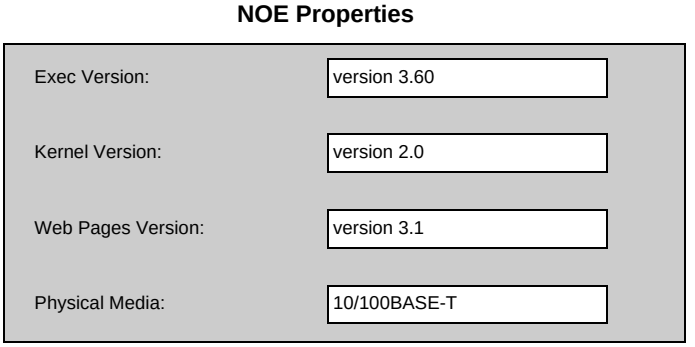
The Crash Diagnostics Page displays a crash log file when a crash has occurred, and a status message when no crash has occurred.

Press **Clear Crash Log File** to clear the crash log file.

NOE Properties Page

Introduction to the NOE Properties Page

You can navigate to the **NOE Properties Page** from the Diagnostics page. (See *Illustration, p. 38*) The **NOE Properties Page** displays the versions of the Exec, Kernel, and Web Pages and the Physical Media:



Copyright 1999-2004, Schneider Automation SAS. All rights reserved.

Note: This page only reports this information. The fields can not be changed.

3.4 I/O Diagnostics

Overview of I/O Diagnostics

Overview

Several Web pages provide information about I/O including:

- configured I/O Remote page,
- remote I/O Drop pages,
- remote I/O Drop Module pages,
- distributed I/O Drops page,
- distributed I/O Specific Drop page,
- distributed I/O Module pages.

Distributed I/O pages can be accessed by selecting either the CPU or a NOM from the Configured Local Rack page assuming that Distributed I/O is configured in the controller.

What's in this Section?

This section contains the following topics:

Topic	Page
Configured Remote I/O Page	52
Remote I/O Drop Pages	54
Remote I/O Drop Module Pages	56
Distributed I/O Drops Page	58
Distributed I/O Specific Drop Page	59
Distributed I/O Module Page	60

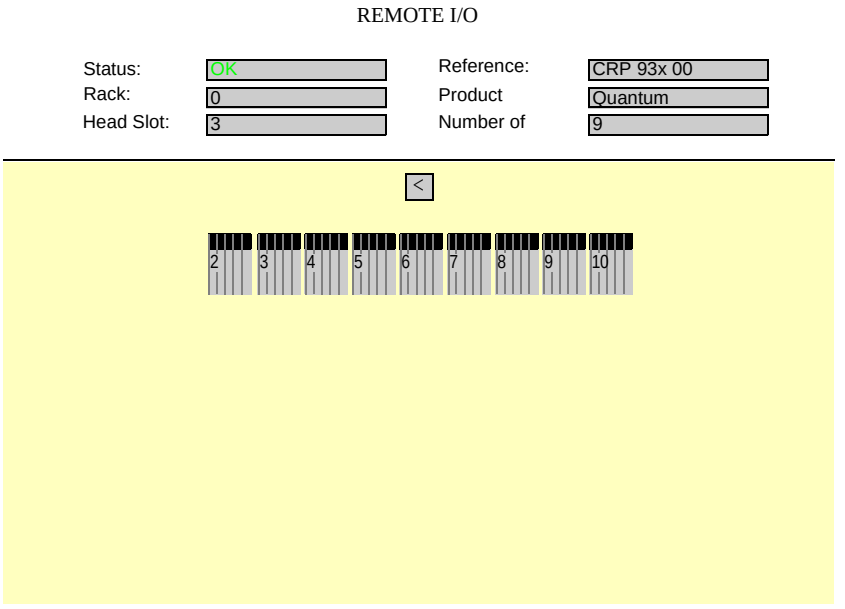
Configured Remote I/O Page

Overview

The Configured Remote I/O page displays information about the Remote I/O Head Processor and the number of remote I/O drops. This page can be accessed by selecting the CRP (RIO Head) module in the Configured Local Rack Page described previously.

Sample Page

Here is an example of a Configured Remote I/O page.



Data

The top half of the screen provides the current status (dynamic) and other data about the Remote I/O Head Processor.

The bottom half of the screen displays an icon for each Remote I/O Drop and the drop number. Moving the cursor across the icons will display a text message in the Browser status window indicating whether the drop is 800 series or Quantum I/O.

Links

Click on a drop adapter icon to get detailed information about each drop.

The "<" back button returns you to the previous page.

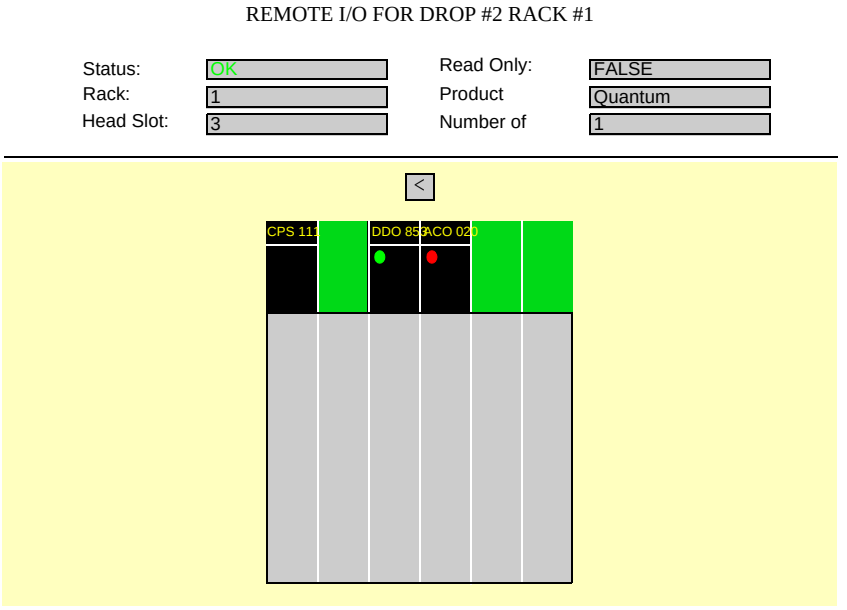
Remote I/O Drop Pages

Overview

When you click the icon for a Remote I/O Drop Adapter on the Configured Remote I/O page, you reach a Remote I/O Drop page with detailed information about that drop.

Sample Page

Here is an example of a Remote I/O Drop page.



Data

The top part of the page reports the current status of the drop adapter and the number of modules in the drop.

The bottom part of the page provides an icon for each module in the drop. A label at the top of the module identifies the module type. Question marks indicated that the module type is unknown or the slot is empty. A colored LED reports module status:

- green indicates that the module is functioning properly,
- red indicates that the module is not functioning properly.

Links	Click a module icon to get detailed information about that module. The "<" back button returns you to the previous page.

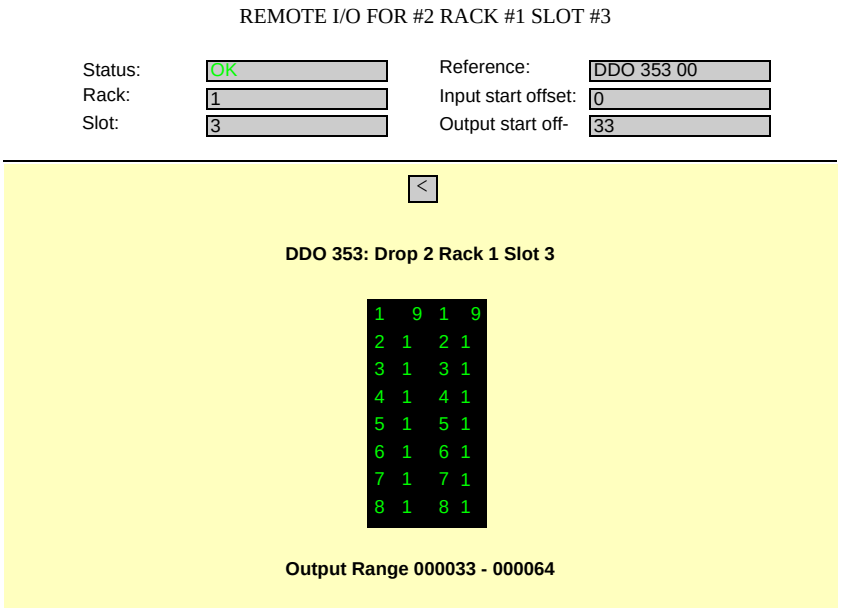
Remote I/O Drop Module Pages

Overview

When you click on a specific module on a Remote I/O Drop page, you reach a Remote I/O Module page with information about that module.

Sample Page

Here is an example of a Remote I/O Discrete Module page.



Data

The top part of the screen provides information about the current status of the I/O module, its location, module type, and input or output offset.

The LED panel in the lower part of the screen displays the status of the discrete I/O points:

- green indicates the point is active,
- off indicates the point is not active,
- it displays analog register values in integer format.

LInks A "<" back button at the bottom of the screen allows you to return to the previous page.

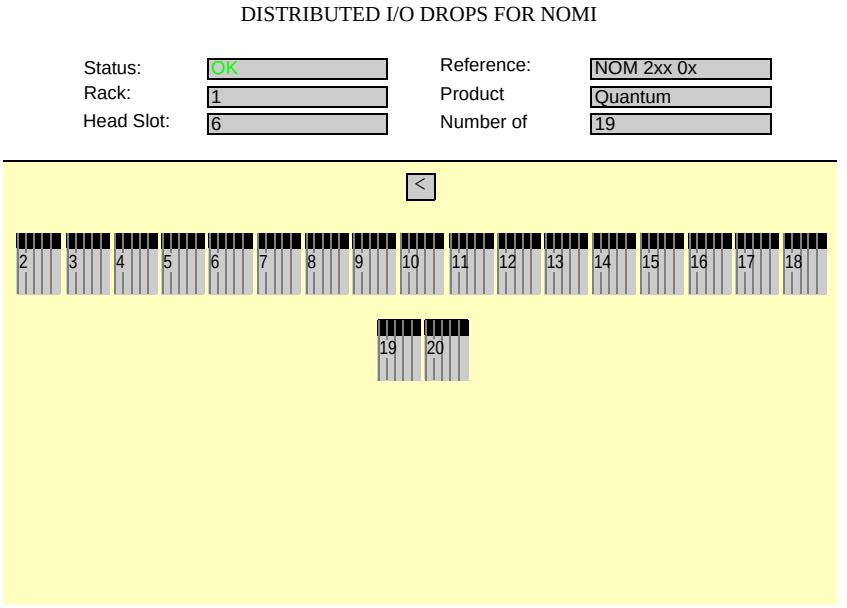
Distributed I/O Drops Page

Overview

When you select a module on the Configured Local Rack page configured for distributed I/O, you reach a page with detailed information about the distributed I/O network drops controlled by the module.

Sample Page

Here is an example of a Distributed I/O drops page connected to the CPU.



Data

The top half of the screen provides the current status (dynamic) and other data about the controller or NOM module running the distributed I/O network.

The bottom half of the screen displays an icon for each distributed I/O drop.

Links

Click a drop icon to get detailed information abut each drop.

The "<" back button returns to the previous page.

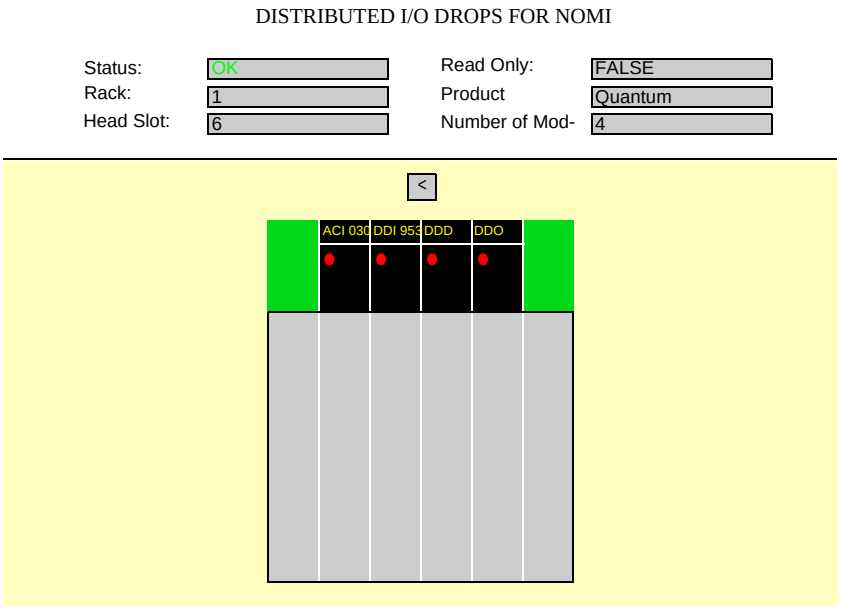
Distributed I/O Specific Drop Page

Overview

When you click a drop icon on the Distributed I/O Network page, you reach a Distributed I/O Drop page with information about that drop.

Sample Page

Here is an example of a Distributed Specific I/O Drop page.



Data

The top part of the page reports the current status of the drop and the number of modules in the drop.

The bottom part of the page provides an icon for each module in the drop. A label at the top of the module identifies the module type.

- Green indicates that the module is functioning properly
- Red indicates that the module is not functioning properly

Links

Click a module icon to get detailed information about that module.

The "<" back button returns to the previous page.

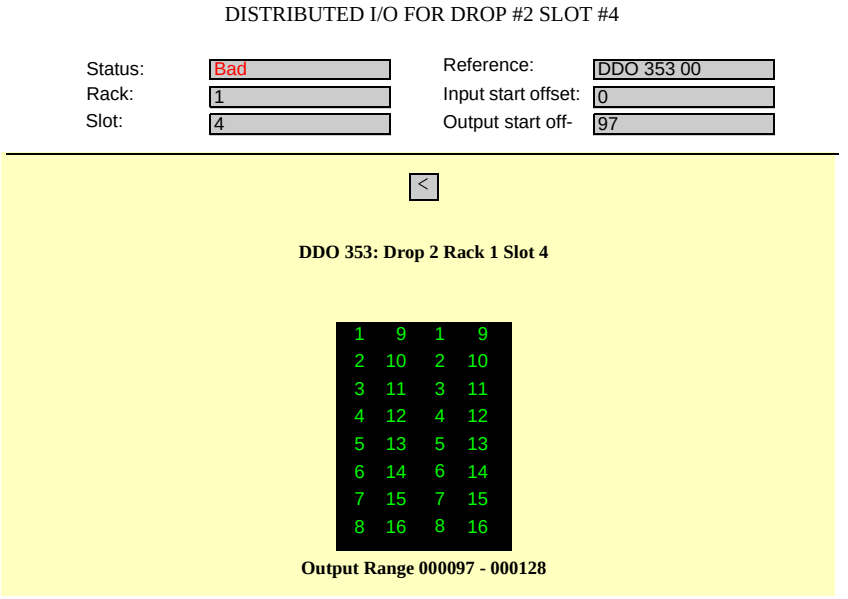
Distributed I/O Module Page

Overview

When you click a module icon on a Distributed I/O Drop page, you reach a Distributed I/O Module page with information about that module.

Sample Page

Here is an example of a Distributed I/O Module page.



Data

The top part of the screen provides information about the current status of the I/O module, its location, module type, and input or output offset.

The LED panel in the lower part of the screen displays the status of the discrete I/O points:

- Green indicates the point is active
- Off indicates the point is not active
- It displays analog register values in integer format

Links

The "<" back button returns you to the previous page.

3.5 Setup Page

Overview of the Setup page

Overview

This section descibes the different configuration services proposed by the ‘Setup’ page.

What's in this Section?

This section contains the following topics:

Topic	Page
Setup Home Page	62
Configure SNMP Page	64
Configuring Faulty Device Replacement	67
Configuring the Global Data (Publish/Subscribe) Utility	71

Setup Home Page

Home Page

The NOE 771 11 **Setup** page page lists the various services used to configure the module. You can navigate to this page from the link on the Welcome Page (See *Home Page, p. 34*).

Illustration

The **Setup** page looks like this:



Setup Page Links These links are on the **Setup** page:

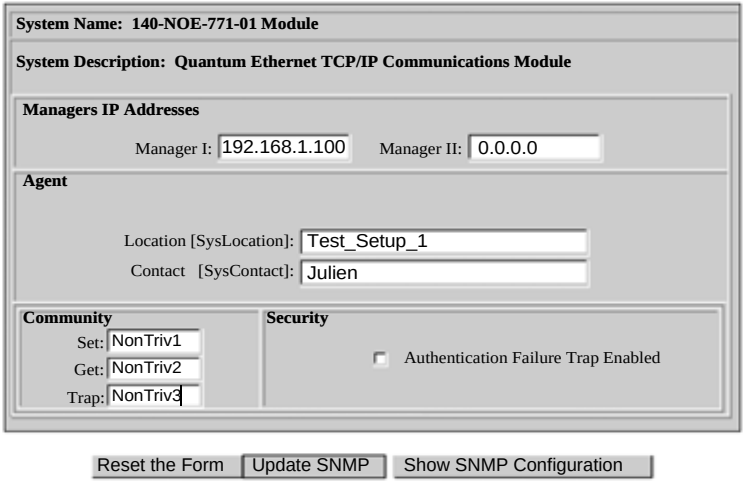
Link	Result
SNMP	Provides the ability to configure the SNMP Agent in the NOE.
Address Server	Configure the IP assignments, including showing the BOOTP and DHCP database.
Global Data	Displays the Global Data Configuration page. Configure the Group Address, Multicast filtering, Distribution period, Health Time Out, Health Bits, and Data Zones. The Global Data Configuration page also displays a Variable Table.
NTP	Configure the IP address of the primary and secondary NTP server. Set a polling period. Select a time zone from the drop-down list, and automatic adjustment for daylight savings time.
Email	Configure the email server's IP and Port (default port is 25). If security required, select Enable and set a Login and Password. Create three mail headers.

Configure SNMP Page

Introduction to the Configure SNMP Page

SNMP may already be configured on your module. If it is not, complete the SNMP Configuration form, which is shown in the following figure.

SNMP Configuration



The following table shows how to enter the required information for configuring SNMP on your module.

Task	How To
To display the current SNMP configuration	Click **Show SNMP Configuration**
To clear the fields	Click **Reset the Form**
To change the SNMP configuration	Change the information on the page and click **Update SNMP**

SNMP Page Fields

The following table describes the specific SNMP fields that you can modify.

Field	Information To Be Supplied
Manager I	IP Address of first SNMP Manager
Manager II	IP Address of second SNMP Manager
Location [SysLocation]	Location of the module
Contact [SysContact]	Name of the responsible systems engineer
Set	Designation of level of user who can set the configuration
Get	Designation of level of user who can view the configuration
Trap	Designation of level of user who can capture information
Authentication Failure Trap Enabled	Turns on Community Name Checking

Completion Message

Clicking on the **Update SNMP** button displays a new page containing the message "Successfully updated SNMP database". Note that this page contains the same links as those on the Configure SNMP Page.

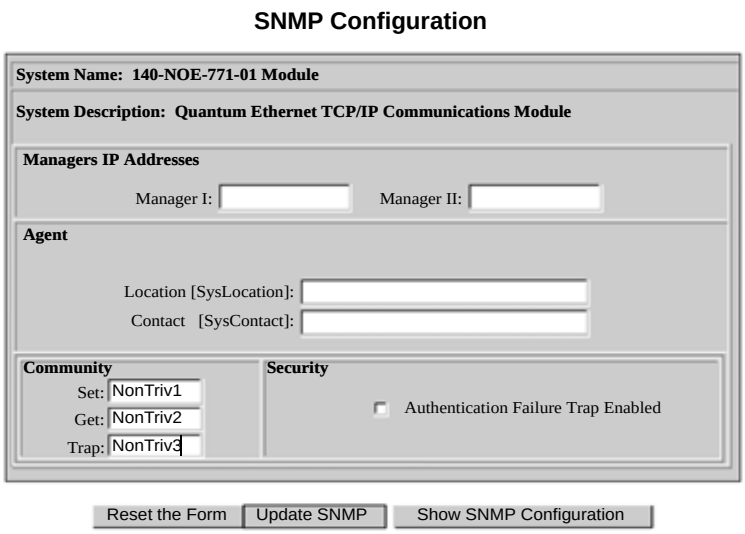
Note: The NOE module has to be reset for the changes to take effect.

SNMP Community Strings

Use strings to restrict access to the SNMP Agent. These strings should be set to Non-Trivial names during module installation.

Modifying the SNMP Community Strings

The following steps should be used to establish the SNMP Community Strings:

Step	Action
1	Enter the URL into your browser: **http://hostname/secure/embedded/builtin?submit=Configure+SNMP** or navigate to the **SNMP Configuration** Web page
2	Enter the **Community** names for **Set**, **Get**, and **Trap** into the **SNMP Configuration** Web page as shown below. 
3	Click **Update SNMP**.
4	After making changes in the **SNMP Configuration** Web page and to set those changes, reboot the module using hot swap.

Configuring Faulty Device Replacement

Configuring the Address Server

To configure the Address Server you use Web pages generated by the embedded Web server. The first page that appears is the **Address Server Configuration** page. The first column contains buttons used for selecting devices. The **Address Server Configuration** page displays configuration information for each device in the system and has seven columns in the table.

Displayed on this page is information about:

- Role Name
- MAC Address
- IP Address
- Subnet Mask
- Gateway

A additional, unnamed column indicates any difference between the current and stored configuration. If a difference exists, an exclamation point is displayed.

This is the **Address Server Configuration** page. All devices are compatible.

Address Server Configuration

	Role Name	MAC Address	IP Address	Subnet Mask	Gateway	
○	myNIP		192.168.3.11	255.255.255.0	192.168.3.11	
○	ENT_1st_floor		192.168.5.14	255.255.0.0	192.168.2.1	

Refresh Address Server Database Table

Add a New Entry | Change an Entry | Delete an Entry

On the Address Server Configuration page you can:

- add a New Entry,
- change an Entry,
- delete an Entry,
- refresh the data table.

Choosing Options

The Address Server Configuration page allows you to choose different options for adding or altering the configurations of your NOE. The options available to you are:

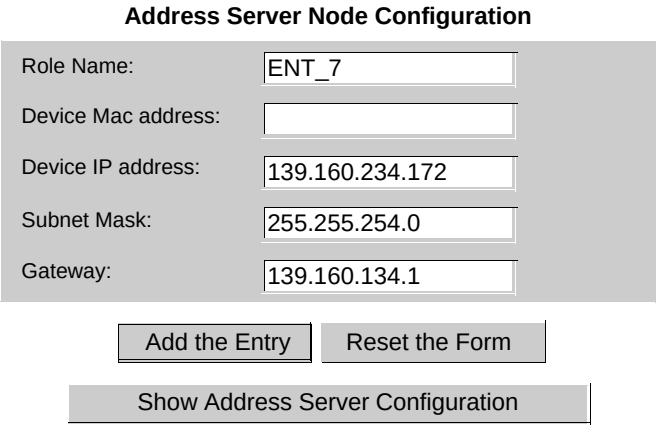
- select an entry
- add an entry
- change an entry
- delete an entry

Below we describe the method and options chosen to perform any of the four options listed above. Screen shots are presented to accompany the *Adding an entry* section.

Selecting an Entry When the page displays, by default no entries are selected. Use the radio buttons in the first column to select an entry. Only one entry may be selected at a time.

Adding an Entry When the **Add a New Entry** button is selected, the **Address Server Node Configuration** page appears. This page displays information about a device.
If you selected a device, this page displays the device's configuration. Configuration information appears in four of the five fields of the dialog box. Only the Role information field is blank. You should enter a Role name, for example *ENT_7*.
If no device is selected, default values appear in all the fields.

Changing an Entry Before using this button, you must select an entry in the database by choosing one of the radio buttons in the first column. If you fail to choose an entry, an error message appears.
When the **Change an Entry** button is selected, the **Address Server Node Configuration** page appears. The information displayed is for the device selected..



Deleting an Entry Before using this button, you must select an entry in the database by choosing one of the radio buttons in the first column. If you fail to choose an entry, an error message appears.

The entry selected will be removed from the database. Before completely deleting an entry, a warning message appears. Click **Yes** if you want to delete the entry, **No** if you don't.

If you click **Yes**, a dialog box appears.



Click **OK**. Another dialog box appears notifying you that the deletion was successful.



Highlighting Errors

If there are problems with the entered configuration parameter information, the system indicates problems using a highlighting mechanism. All the configurations appear in purple and italic, and the device with configuration problems appears in red and bold.

The system detects the following errors:

- Bad **Role Name**,
  The valid **Role Name** characters are:
  - a through z (lower case),
  - A through Z (upper case),
  - "_" (underscore),
- Bad **MAC Address**,
  The valid **MAC Address** characters are:
  - 0 through 9,
  - A through F,
- Wrong **IP Address**,
- Wrong **Subnet Mask**,
- Wrong **Gateway**,
- Double Entry.
  Each entry must have a unique **Role Name** or **MAC Address**. If a duplicate **Role Name** or **MAC Address** is entered, the system highlights the error.

Erroneous entries are not loaded into the DHCP server. Therefore, errors must be corrected before loading. There are two ways of correcting the error:

- **Correcting through Web page:** Make the changes on the Web page and submit the change.
- **Correcting through the Address Server configuration file:** Make the changes in the file and reboot the server.

Configuring the Global Data (Publish/Subscribe) Utility

Overview

Whether you use either the Configure Each Device Separately or the Copy Configuration method, the procedure to configure individual parameters is the same. Therefore, in order to use the Global Data (publish/subscribe) utility in the NOE, you need to configure the Global Data parameters including:

- distribution period,
- multicast filtering,
- health bit location,
- global Data base address,
- group IP address.

The following sections describe in detail the exact steps to configure each parameter on the **Global Data Configuration** page.

Illustration

You can change the configuration in the **Global Data Configuration** page:

Global Data Configuration

Group Address 239 . 200 . 255 . 255 . ☑ Multicast filtering | Distribution period 5 scan

Health Time Out 1000 ms | Health Bits %MW 4 to 7 | Data Zone %MW 0 to 5

Update Global Data Configuration

Variable Table

Data	Type	Symbol	Address	Length
1	SUB	var_01	%M 48	2
2	PUB	var_02	%M 60	2
3	SUB	var_03	%M 44	2
4	NON		%M	
5	NON		%M	
6	NON		%M	
7	NON		%M	
8	NON		%M	
9	NON		%M	

Configuring Global Data

After you have completed the Modeling System Configuration process using the second method, Copy Configuration, then you modify the following parameters:

- distribution period,
- health Time Out,
- health Bits location,
- start address,
- type: Pub / Sub / None.

Please **do not** change Symbol (description), and Length.

To change the Global Data variables of the group box on the **Global Data Configuration** page, follow the instructions below.

Step	Action
1	Adjust the **Distribution Period Cycle**. Enter a value from 1 through 50. **Note: Distribution period** is the minimum number of controller scan times before an update will occur.
2	Before entering a value in the **Group address** field, identify the station's distribution group. The **Group address** entry will be an IP address from **`224.0.0.0`** through **`239.255.255.255`**. **Group address:** the Class D Multicast IP address used for a distribution group. All members of this distribution group are configured to use the same group address, and therefore, all members can communicate with each other using Global Data.
3	Set the timeout in the **Health Time Out** field. This value is measured in milliseconds and can be set to a value that ranges from 50 through 15000 ms (in 50 ms increments) **Note: Validity Time** is the maximum time between received subscriptions before a subscription is declared unhealthy (faulty).
4	In the **4x Starting Address** set the **Data Zone** field.
5	If you are connected to an Ethernet switch that supports multicast filtering, click the **Multicast filtering** check box.
6	Enter %MW word (4x register) location for the Health Bits. This is the location for storing health bits.

Note: Health bits run in different directions.

- I/O scanner health bits run left to right.
- Global Data health bits run right to left.

Changing Global Data Variables

To change the Global Data variables that appear in the **Variable Table** area, follow the instructions below.

Step	Action
1	Highlight the identification number in the **Data ID** column.
2	In the **Type** column select the publish/subscribe variable type from the drop down list. Three options are available publish, subscribe, or none, displayed as: • **NONE** • **SUB** • **PUB**
3	In the **Symbol** column you may enter text to describe the variable.
4	In the **Address** column you see the application address for this variable. **Note:** This is a read only field.
5	In the Length column for each row, type a value, which represents the number of 4x registers. The ending 4x register field is automatically updated. If you are using the second method, **Copy Configuration**, update **Length** the first time only.
6	When you are finished, click the **Update Global Data Configuration** button.

Verifying System Operation

To ensure that the system is operational, do the following:

Step	Action
1	Verify that all controllers are running.
2	Look at the health of all variables using the **Global Data Diagnostics** page. Follow these links: | **Diagnostics** | **NOE Diagnostics** | **Global Data**

Default Web Site for Premium

4

Default Web Site for Premium

Overview

When you receive the PLC module, it already contains a default Web site with the Rack Viewer and the Run-Time Data Editor pages pre-loaded.

You may view these pages simply by installing the module and configuring its IP address. To access the site, type the IP address of the module in your browser and enter the default user name and password of "USER". However, Schneider Electric recommends that you complete the setup procedures as outlined in *Creating a New Configuration, p. 147*.

This section describes the pages in the default Web site.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
4.1	Premium Home Page	76
4.2	Monitoring Page	77
4.3	Diagnostics Page	79
4.4	Option Modules Diagnostics	91
4.5	Setup Page	107

4.1 Premium Home Page

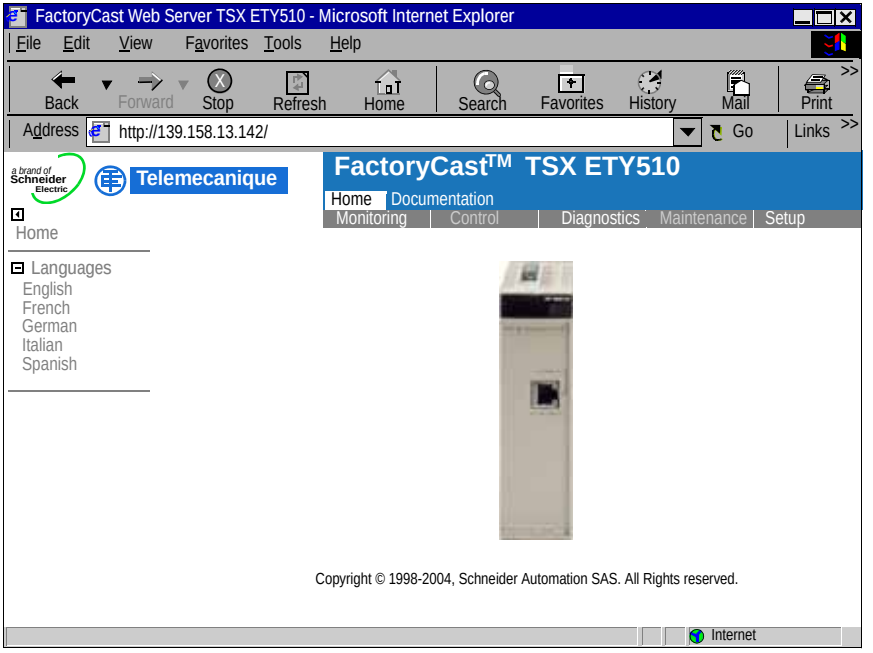
Premium Home Page

Overview

The visitor accesses the Premium home page by entering the IP address of the module in his web browser. No password is required to display this page.

Home Page

This Premium home page looks like this:



Links

From the Premium home page, you can access the following pages :

- Monitoring (See *Monitoring Home Page, p. 77*),
- Diagnostics (See *Diagnostics Home Page, p. 80*),
- Setup (See *Setup Home Page, p. 108*),
- Documentation,
- Foreign language links exist for French, German, Italian and Spanish versions.

The visitor will have to supply a user name and a password to access the services proposed in these pages.

4.2 Monitoring Page

Monitoring Home Page

Home Page	This page lists the various viewer services supported by the default web site of the module and provides links for accessing the services you require.

Illustration

The ‘Monitoring’ home page looks like this:



To access the service you require, click on a link.

- Data Editor (See *Data Editor, p. 192*): for creating variable data tables, so as to be able to determine their value when the table is animated.
- Data Editor Lite (See *Data Editor Lite, p. 209*): for creating variable data tables, so as to be able to determine their value when the table is animated (lighter than Data editor).
- Graphic Editor (See *Graphic Editor, p. 220*): for creating graphics, so as to be able to determine the value of variables when the graphic is animated.
- Graphic Viewer (See *Graphic Viewer, p. 268*): for viewing graphics, so as to be able to determine the value of variables when the graphic is animated.
- Password-protected custom pages (See *Adding Custom Pages to the Site, p. 271*): for viewing screen pages (accessible with password) created by the user
- Custom pages without password protection (See *Adding Custom Pages to the Site, p. 271*): for viewing screen pages (accessible to everyone) created by the user.

4.3 Diagnostics Page

Overwiew of the Diagnostics Page

Overview

This section describes the different services proposed by the ‘Diagnostics’ page.

What's in this Section?

This section contains the following topics:

Topic	Page
Diagnostics Home Page	80
Rack Viewer Page	81
PLC Personality Page	82
Ethernet Module Statistics Page	83

Diagnostics Home Page

Home Page

This page lists the various services supported by the default web site of the module and provides links for accessing the services you require.

Illustration

The Diagnostics home page looks like this:



Links

To access the service you require, click on a link:

- Rack Viewer,
- Alarm viewer (See *Alarm Viewer, p. 337*),
- Ethernet.

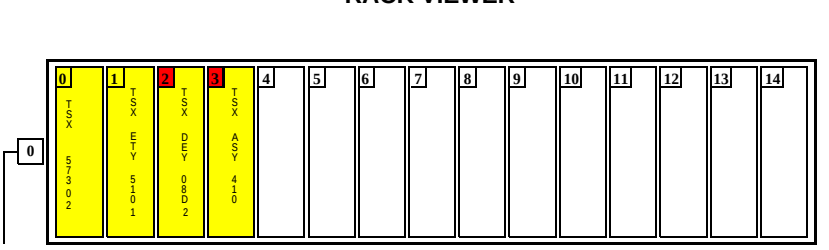
Rack Viewer Page

Overview

The Rack Viewer Page displays the current configuration of the racks (local or remote), including the controller, Embedded Server module, and any I/O modules.

Sample Page

Here is an example of a Rack Viewer page.



Data

The following information is provided for each module displayed in the rack.

- A vertical label displays the module type and part number
- The box in the upper left hand corner of the module displays the slot number and module health:
  - A yellow box indicates that the module is functioning properly
  - A red box indicates that the module is not functioning properly

Links

Click a module icon to obtain detailed information about that module.

The CPU module icon (slot 1) contains two links. The top link leads to the FIP I/O Module Diagnostics page. The tower link leads to the PLC Personality page.

A line leading down from the Rack#0 icon (to the left of the rack) is a link to the next rack. When you place your mouse over this link, it turns into a red arrow. Each rack in the configuration may be viewed in turn. Upward links will take you back toward Rack#0.

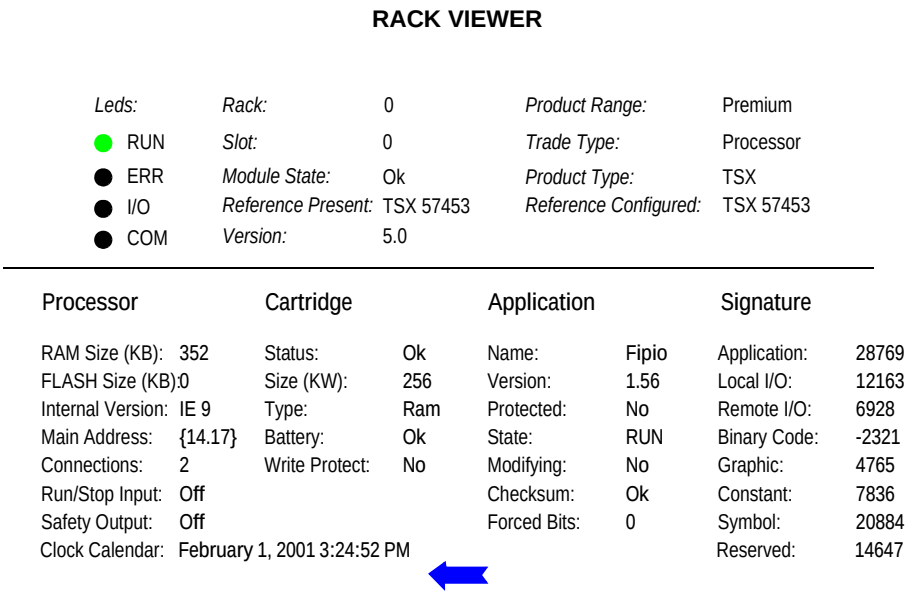
PLC Personality Page

Overview

The PLC Personality page provides information about the controller and its configuration.

Sample Page

Here is an example of a PLC Personality page.



Dynamic Data

The LEDs in the upper left hand corner of the screen provide a dynamic report on the controller status.

LEDs	Color if On	Meaning if On	Meaning if Blinking	Meaning if Off
RUN	Green	Application running	Stopped	PLC error
ERR	Red	PLC error	Not configured	No error
I/O	Red	I/O event	---	No error
COM	Yellow	Communication error	---	No error

Links

The back arrow will take you to the Rack Viewer page for this controller.

Ethernet Module Statistics Page

Home Page

The ‘Ethernet’ menu contains a list of links for accessing the different diagnostic pages for the Ethernet module:

- Global Data utility,
- I/O scanning utility (See *I/O Scanning Page, p. 85*),
- Messaging utility,
- Bandwidth monitoring utility,
- Ethernet module statics,
- NTP utility,
- Email utility.

A link also allows the downloading of the private MIB source file.

Click on a link to access the desired diagnostics page.

Global Data Page

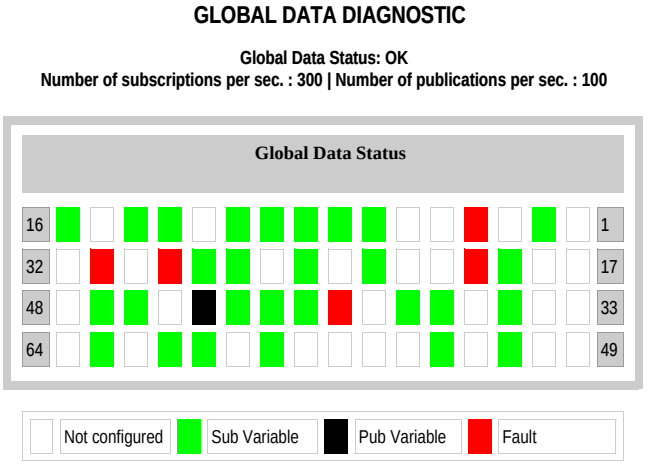
Information on the general diagnostics of Global Data can be found above this page:

- Status,
- Number of publications per second,
- Number of subscriptions per second.

This page also shows a table that regroups all published and subscribed variables in the same distribution group. The nature of each variable is identified by its color code:

- **green:** subscribed variables,
- **black:** published variables,
- **white:** unconfigured variables,
- **red:** variables with communication faults.

View of the Global Data diagnostics page:



I/O Scanning Page

Information on the general diagnostics of the I/O scanning utility can be found above this page:

- Status,
- Number of transactions per second,
- Number of connections per second.

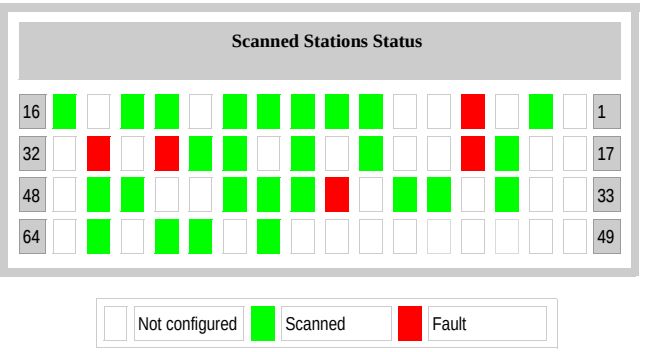
This page also displays a summary of the status of all modules:

- Green for the **scanned** modules,
- White for the **unconfigured** modules,
- Red for **faulty** modules,
- Black for the modules which are temporarily **unscanned**.

View of the I/O scanning diagnostics page:

I/O SCANNING DIAGNOSTIC

I/O Scanning Status: OK
Number of transactions per sec. : 1000 | Number of connections: 20



Messaging Page

The Messaging page provides current information on the open TCP connection on port 502.

MESSAGING DIAGNOSTIC

Number of Messages sent: 150 | Number of Messages received: 50

Conn.#	Remote addr.	Remote port	Local Port	Mess. sent	Mess. received	Err. sent.
1	192.160.10.20	1920	502	20	12	0
2	139.160.235.90	2020	502	0	30	02
3	192.160.10.21	502	3000	3	60	0
4	139.160.234.20	1050	502	15	42	0
5	139.160.234.18	5120	502	0	39	1

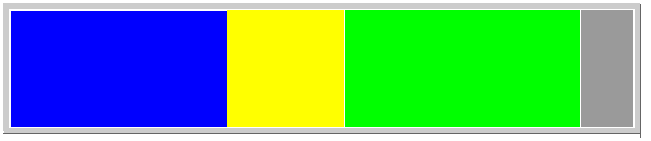
The number of sent/received messages on the port can be found at the top of this page. A table provides, for each connection (numbered from 1 to 64):

- **Remote addr.:** remote IP Address,
- **Remote port:** remote TCP port,
- **Local Port:** local TCP port,
- **Mess. sent:** number of messages sent from this connection,
- **Mess. received:** number of messages received from this connection,
- **Err. sent:** error number on this connection.

Bandwidth Monitoring Page

The bandwidth monitor page shows the load distribution of the TSX ETY 4103/5103 module between the Global Data utilities, I/O Scanning, Messaging, and other utilities:

BANDWIDTH MONITOR



Global data: 30% | I/O Scanner: 20% | Messaging: 40% | Others: 10%

Statistics Page

If you click the Embedded Server module in the Rack Viewer, you will reach the Ethernet Module Statistics page. This page provides up-to-date information about the status, configuration, and activity of the Embedded Server module.

Here is an example of an Ethernet Module Statistics page.

Leds:	*Rack:*	0	*Product Range:*	Premium
RUN	*Slot:*	2	*Trade Type:*	Communication
ERR	*Module State:*	Ok	*Product Type:*	Ethernet
STS	*Reference Present:*	TSX ETY 510	*Reference Configured:*	TSX ETY 510
	Version:	1.1		

Configuration		Activity	
Local IP Address:	139.158.12.110	TCP Messaging connections:	1
Subnetwork Mask:	255.255.218.0	Sent Messages:	485851
Gateway Address:	139.158.8.1	Received Messages:	485790
X-WAY Address:	{0,0}	Refused Messages:	0
X-WAY Bridge:	No	IO Scanning (Msg/s):	3
IO Scanner Connections:	0		

Back

Dynamic Data

The LEDs in the upper left hand corner of the screen provide a dynamic report on the Embedded Server module status.

LEDs	Color if On	Meaning if On	Meaning if Blinking	Meaning if Off
RUN	Green	Running normally	---	Power Off
ERR	Red	Module fault	Not configured	Running normally
STS	Red	Network address fault or station out of range	---	No error

Links

The back arrow will take you to the Rack Viewer page for this controller.

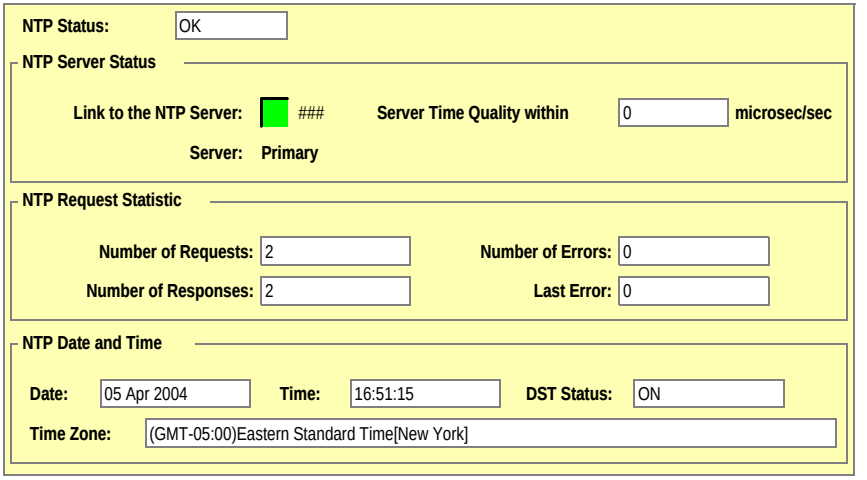
NTP Diagnostics Page

Time synchronization service parameters:

Parameter	Description
NTP status	Service is correctly configured (OK)
NTP server status	NTP client is connected to the NTP server, and if the server is Primary or Standby
NTP requests	Total number of client requests sent to the NTP server
NTP responses	Total number of server responses sent from the NTP server
Number of errors	Total number of unanswered NTP requests
Last error code	Last error code received from the NTP client
Date	Date in d/m/y format
Time	Time
Time zone	Time zone plus or minus Universal Time, Coordinated (UTC)
DST	Daylight saving time (DST) parameter is either **1.** on (enabled) **2.** off (disabled)

The dialog:

NTP DIAGNOSTICS



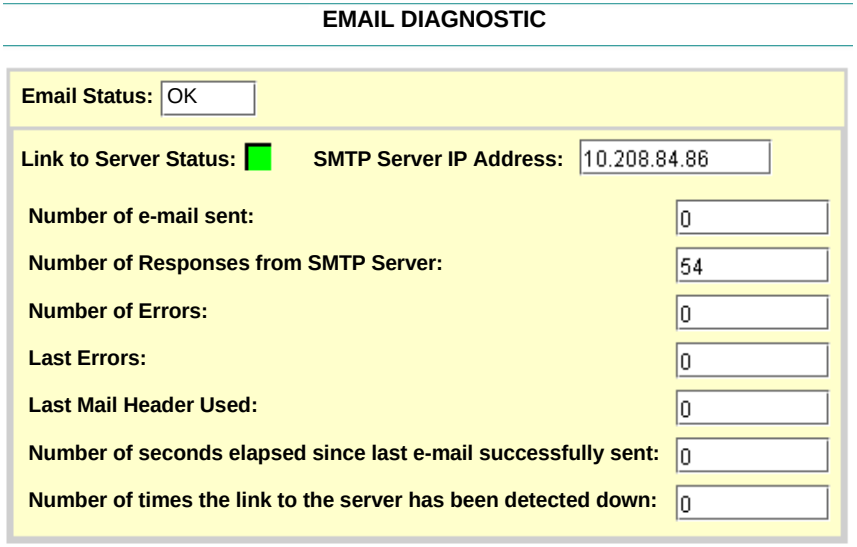
Last Error field displays values, which indicate the type of error.

Type of Error	Value
Component OK and running	0
Excessive network traffic or server overload	1

Type of Error	Value
Bad parameters in the configuration	3
Component is disabled	4
Incorrect IP	9
Time zone file absent	14
Syntax error in the customrules file	15

Email Diagnostics Page

The dialog



Electronic mail notification service parameters

Parameter	Description
Email status	Email service is correctly configured (OK).
Link to Server Status	Ethernet module is connected to the SMTP server Status is checked at start-up and at least every 30 minutes after start-up: ● Green = module connected to server ● Red = module NOT connected to server
SMTP Server IP Address	IP address of the SMTP server
Number of e-mails sent	Total number of emails sent successfully
Number of Responses from SMTP Server	Total number of SMTP messages received from the SMTP server
Number of Errors	Total number of e-mails NOT sent because of an error
Last Errors	Reason for the last error with a code in hexadecimal. 0 displays if no error occurs
Last Mail Header Used	Last header used by giving the number.
Number of seconds elapsed since last e-mail successfully sent	Counts the number of seconds since the last email was successfully sent.
Number of times the link to the server has been detected down	Number of times the SMTP server could not be reached. (Link checked every 30 minutes.)

4.4 Option Modules Diagnostics

Overview of Option Modules Diagnostics

Overview

Several default Web pages provide information about configured option modules

- FIP I/O Module Diagnostics page,
- Digital I/O Module Diagnostics page,
- Analog I/O Module Diagnostics page,
- Communication Module Diagnostics page,
- PCMCIA Communication Diagnostics page,
- Standard Module Diagnostics page.

What's in this Section?

This section contains the following topics:

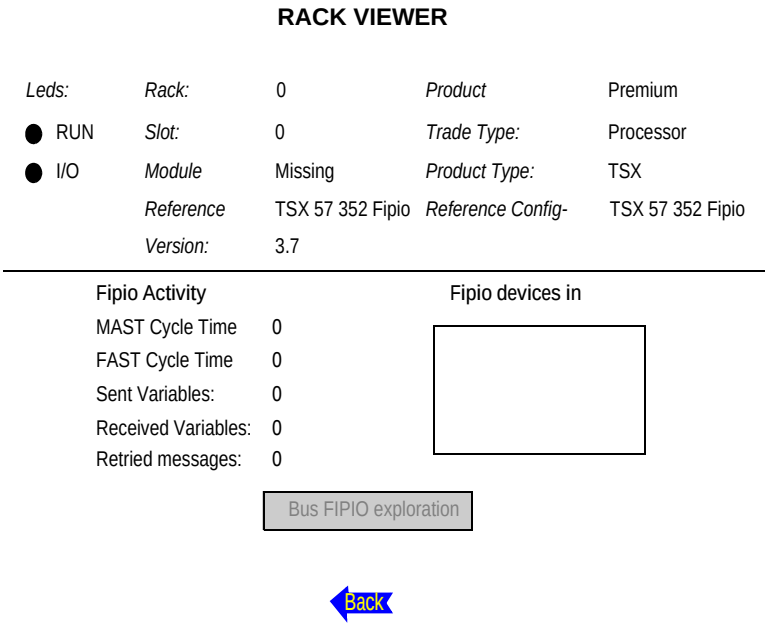
Topic	Page
FIP I/O Module Diagnostics Page	92
Digital I/O Module Diagnostics Page	98
Analog I/O Module Diagnostics Page	100
AS-i Module Diagnostics Page	102
PCMCIA Communication Card Diagnostics Page	104
Standard Module Diagnostics Page	106

FIP I/O Module Diagnostics Page

Overview

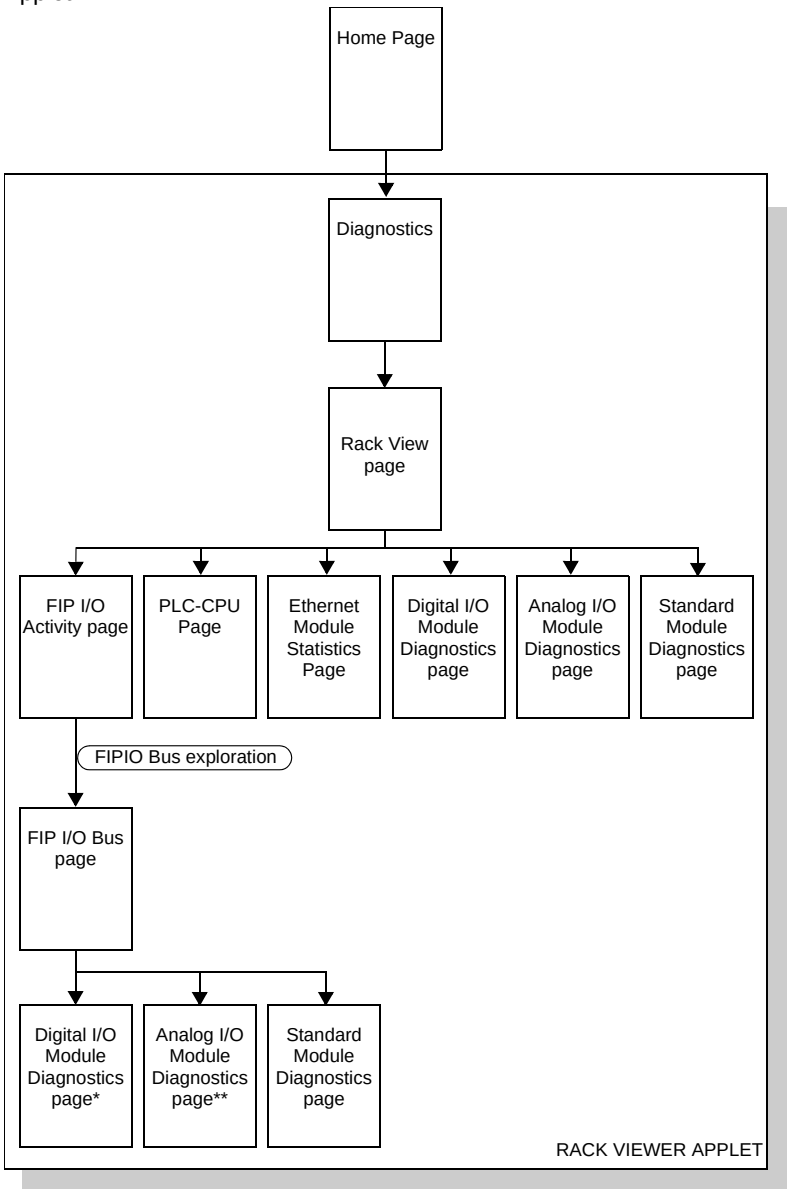
If you click the FIP I/O link on the controller icon in the Rack Viewer page, you will reach the FIP I/O Module Diagnostics page.

This is the FIP I/O page.



Rack Viewer Navigation

The following illustration represents the navigation page tree of the Rack Viewer Applet.

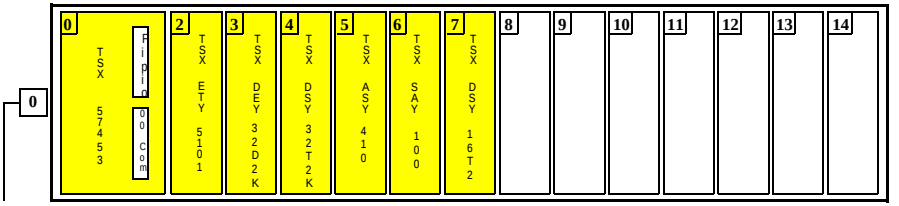


* Only for Digital TBX-7 modules
** Only for Analog TBX-7 modules

Rack Viewer

Click the Rack Viewer link. The applet starts and displays the current configuration of the local rack, including the controller, Embedded Server module, and any I/O modules.

RACK VIEWER



FIP bus link

If Premium has a CPU with FIP bus link, click the FIP I/O link and the first level diagnostic page of FIP I/O is displayed.

RACK VIEWER

Leds:
RUN
I/O

Rack:	0	*Product Range:*	Premium
Slot:	0	*Trade Type:*	Processor
Module State:	Run	*Product Type:*	TSX
Reference Present:	TSX 57352 Fipio	*Reference Configured*	TSX 57352 Fipio
Version:	3.4		

Fipio Activity

MAST Cycle Time (ms):	**6**
FAST Cycle Time (ms):	**0**
Sent Variables:	**33293**
Received Variables**:**	**52412**
Retried messages:	**0**

Fipio devices in fault

Device at connection point 2

Bus FIPIO exploration

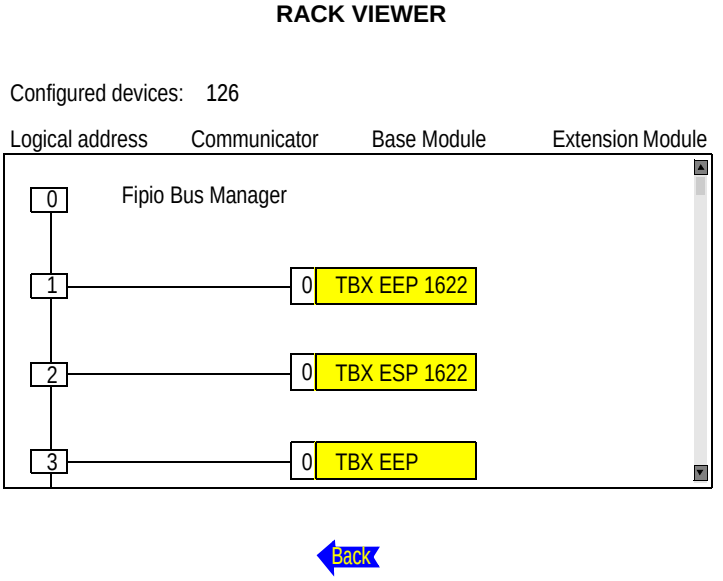
Back

FIP I/O Bus Page

In FactoryCast Client, there is a button at the bottom of the FIP I/O activity page, named *Bus FIP I/O exploration*.If you click on this button, the *FIP I/O Bus page* is displayed.

Note: The button FIP I/O Bus exploration is enable only if a FIP I/O bus is configured in the PLC application. Otherwise, the button is disabled (grayed).

This is the FIP I/O Bus Page.



At the top of the page, the number of FIP I/O devices configured on FIP I/O Bus is displayed. A scroll window displays all configured devices on FIP I/O Bus.

A small circle indicates the device connection point. If a FIP I/O Device is fault, the background color of its logical address indicator is red. A FIP I/O device module in fault is also displayed when the box containing the module number has a red background color.

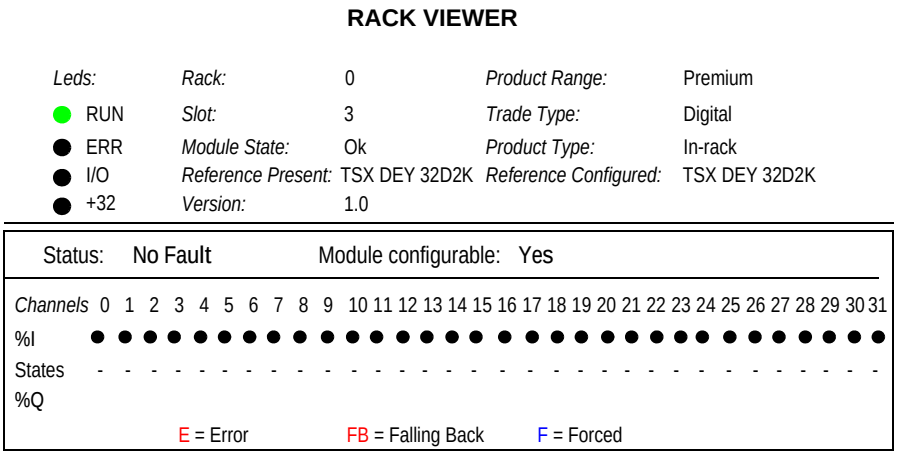
Note: A FIP I/O device displayed in fault by FIP I/O Bus page is also displayed in fault by FIP I/O devices in the fault list of FIP I/O activity page.

If the Base Module of FIP I/O device is "modular" type, a **Communicator** is displayed. If Base Module of FIP I/O device is "compact" type, there is no **Communicator**.

Base Module links and Extension Module links Diagnostic Page

If present, Base Module links and Extension Module links are mouse sensitive

Clicking the module allows you to access its diagnostic page in the same way as an in-rack module.



Types of Diagnostic Pages

On FIP I/O modules, there are three kinds of diagnostic pages. The display depends on Product Range and of Trade Type.

A *Standard Module Diagnostics page* is displayed except for Digital TBX-7 and for Analog TBS-7 modules.

They are displayed respectively as the *Digital I/O Module Diagnostics page* and the *Analog I/O Module Diagnostics page*.

Dynamic Data

The LEDs in the upper left hand corner of the screen provide a dynamic report on the FIP I/O module status.

LEDs	Color if On	Meaning if On	Meaning if Off
RUN	Green	Link active	Link inactive
I/O	Red	Remote device in fault	Remote device operating normally

Links

The back arrow will take you to the Rack Viewer page for this controller.

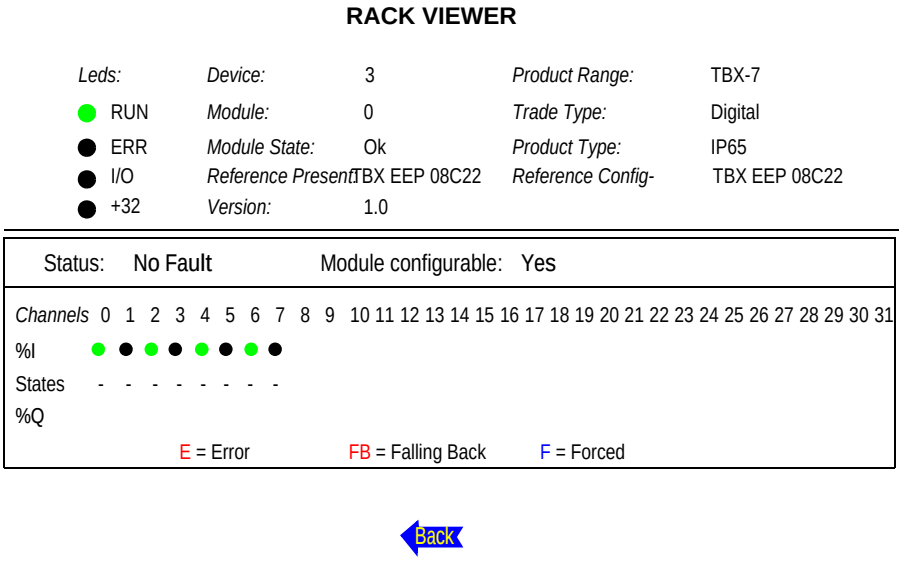
Digital I/O Module Diagnostics Page

Overview

If you click a digital I/O module in the Rack Viewer, you will reach a Digital I/O Module Diagnostics page with detailed information about that module.

Sample Page

Here is an example of Digital I/O Module Diagnostics page.



Dynamic Data

The LEDs in the upper left hand corner of the screen provide a dynamic report on the module status.

LED	Color if On	Meaning if On	Meaning if Blinking	Meaning if Off
RUN	Green	Running normally	---	Faulty module
ERR	Red	Module failure	Communication fault	No error
I/O	Red	Overloaded short circuit or server/preactuator voltage fault	Terminal Block fault	No error
+32	Green	Channels 32 ... 63 displayed	---	Channels 0 ... 31 displayed

The LED panel in the lower part of the screen provides a dynamic status report for each channel.

Links

The back arrow will take you to the Rack Viewer page for this controller.

Analog I/O Module Diagnostics Page

Overview

If you click an analog I/O module icon in the Rack Viewer, you will reach an Analog I/O Module Diagnostics page with detailed information about that module.

Sample Page

Here is an example of an Analog i/O Module Diagnostics page.

RACK VIEWER

Leds:	*Rack:*	0	*Product Range:*	Premium
RUN	*Slot:*	5	*Trade Type:*	Analog
ERR	*Module State:*	Ok	*Product Type:*	In-rack
I/O	*Reference Present:*	TSX ASY 410	*Reference Config-*	TSX ASY 410
	Version:	1.0		

Analog Channel Viewer

Channels	*Channel 0*	*Channel 1*	*Channel 2*	*Channel 3*	
%IW					
States					
%QW	10000	10000	10000	10000	
			E = Error		F = Forced

Back

Dynamic Data

The LEDs in the upper left hand corner of the screen provide a dynamic report on the module status.

LED	Color if On	Meaning if On	Meaning if Blinking	Meaning if Off
RUN	Green	Running normally	---	Faulty module or no power
ERR	Red	Module failure	Communication fault with PLC	No error
I/O	Red	Range overshoot or sensor link fault	Terminal Block fault	No error

Links

The back arrow will take you to the Rack Viewer page for this controller.

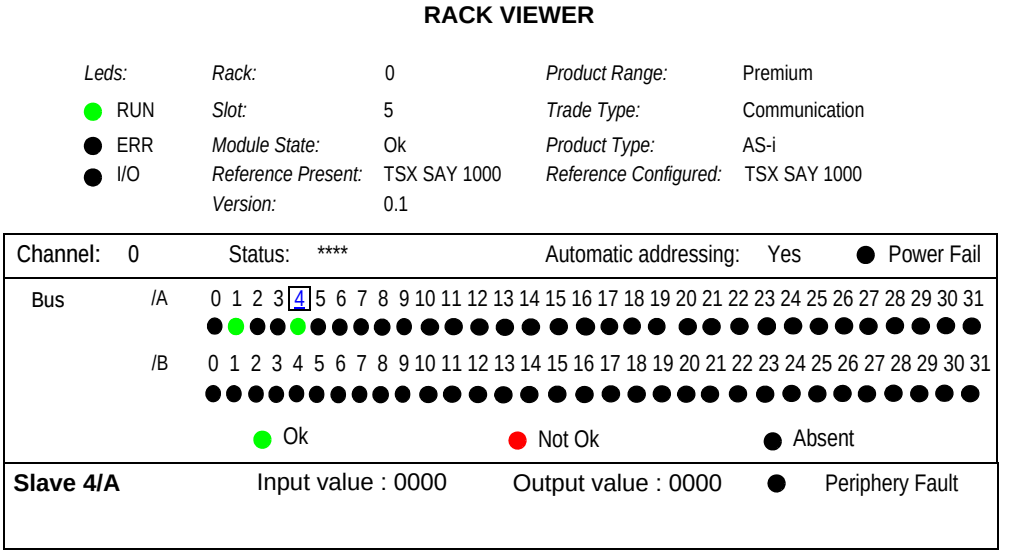
AS-i Module Diagnostics Page

Overview

If you click an AS-i module icon in the Rack Viewer, you will reach an AS-i Module Diagnostics page with detailed information about that module.

Sample Page

Here is an example of an AS-i Module Diagnostics page.



Note: Clic on slave green LED to display the slave I/O values. When a slave is selected, I/O values and "Peripheral Fault" LED are displayed.

Dynamic Data

The LEDs in the upper left hand corner of the screen provide a dynamic report on the module status.

LED	Color if On	Meaning if On	Meaning if blinking	Meaning if Off
RUN	Green	Module **OK** and configured	Awaiting configuration	HS Module or autotest fault
ERR	Red	Serious non-rectifiable Module fault	Rectifiable module fault (PL7 configuration, AS-i power supply)	Module **OK**
I/O	Red	AS-i bus Fault	Fault or awaiting user configuration	AS-i Bus **OK**

Links

The back arrow will take you to the Rack Viewer page for this controller.

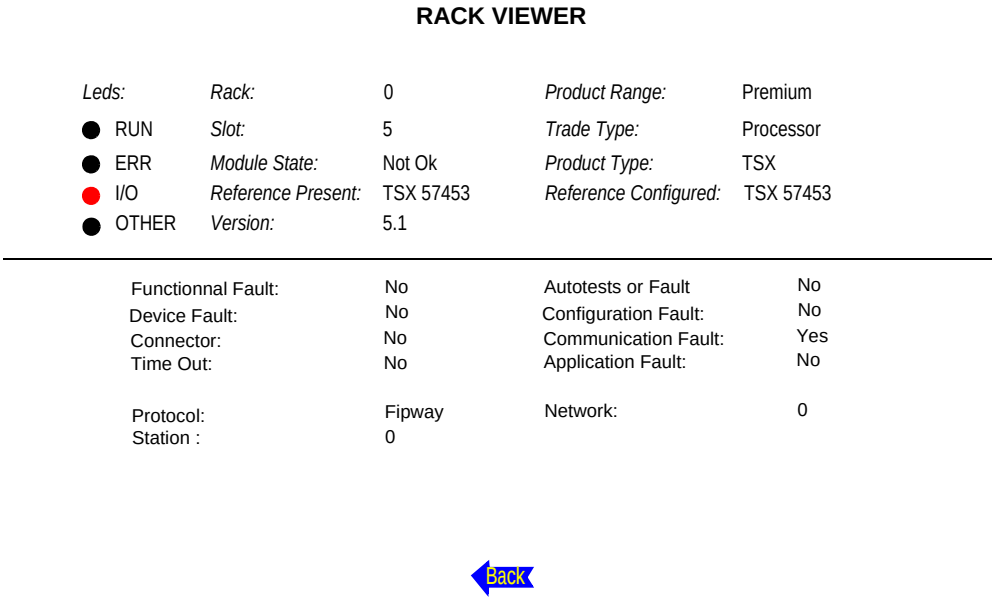
PCMCIA Communication Card Diagnostics Page

Overview

If you click a PCMCIA icon in the Rack Viewer, you will reach a PCMCIA Communication Card Diagnostics page with detailed information about that module.

Sample Page

Here is an example of a PCMCIA Communication Card Diagnostics page.



Dynamic Data

The following table describes the role of each LED.

LED	Color if On	Meaning if On	Meaning if Flashing	Meaning if Off
RUN	Green	PLC running normally, program executing.	PLC in STOP mode or blocked by software error.	PLC not configured: application missing, invalid or incompatible.
ERR	Red	processor or system error.	• PLC not configured (application missing, invalid or incompatible), • PLC blocked by a software error, • memory card battery error, • X Bus error.	normal state, no internal error.
I/O	Red	input/output errors coming from a module, a channel or a configuration error.	X Bus error.	normal state, no internal error.

Links

The back arrow will take you to the Rack Viewer page for this controller.

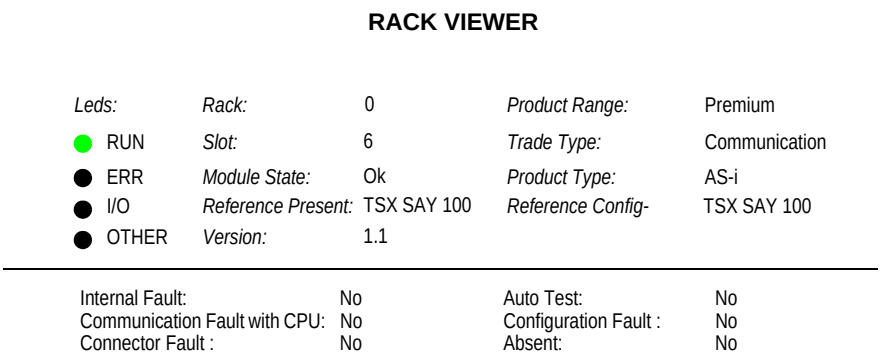
Standard Module Diagnostics Page

Overview

If you click any other type of module in the Rack Viewer page, you reach a standard Module Diagnostics page with detailed information about that module.

Sample Page

Here is an example of a Standard Module Diagnostics page.



Dynamic Data

The LEDs in the upper left hand corner of the screen provide a dynamic report on the module status.

LED	Color if On	Meaning if On	Meaning if Blinking	Meaning if Off
RUN	Green	Running normally	***	***
ERR	Red	Module fault	Not configured	No error
I/O	Red	I/O event	---	No error
OTHER	Yellow	***	***	***
*** The meaning depends on the module type. For more information, refer to the user manual for the appropriate module.				

Links

The back arrow will take you to the Rack Viewer page for this controller.

4.5 Setup Page

Overview of the Setup page

Overview

This section descibes the different configuration services proposed by the ‘Setup’ page.

What's in this Section?

This section contains the following topics:

Topic	Page
Setup Home Page	108
Address Server Page for the HTTP Server	109

Setup Home Page

Home Page

This page lists the various services used to configure the module.

Illustration

The ‘Setup’ home page looks like this :



Links

To access the configuration service you require, click on a link :

- ‘Address Server’,
- ‘NTP’ (See *Configuring the Time Synchronization Service, p. 116*),
- ‘Email’ (See *Configuring the Electronic Mail Notification Service, p. 114*).

Address Server Page for the HTTP Server

At a Glance

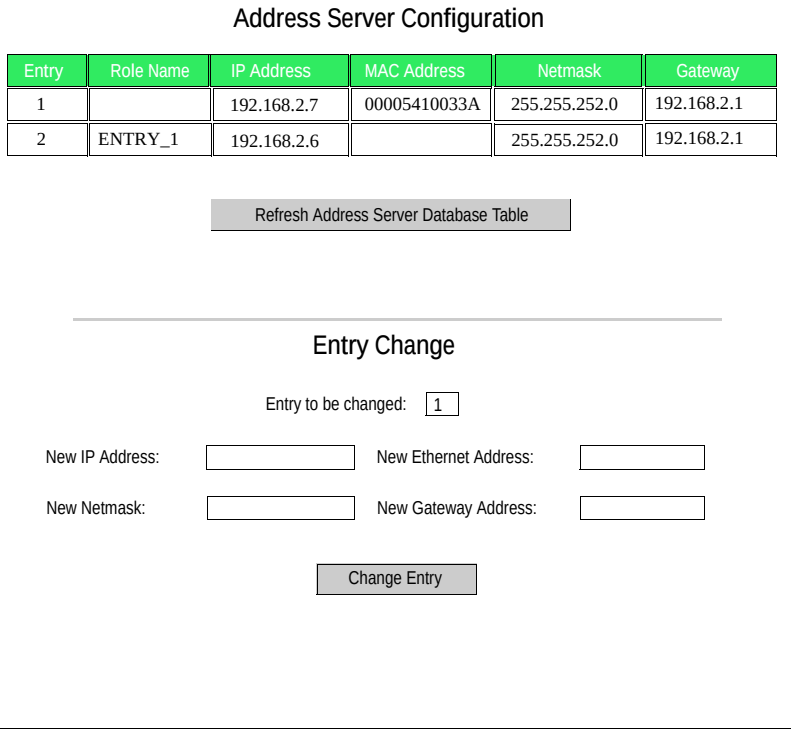
This page is used to display or modify the correspondence table between the MAC addresses or the Name (Role Name) and the IP addresses of the module if the latter is configured as a BOOTP server.

This function is useful when replacing a failed remote device (e.g. replacing a faulty Momentum module).

Note: This page does not allow the addition of new inputs, nor the modification of the Name (Role name) for a remote device.

Illustration

By way of example, the Address Server page of a TSX ETY4103, a TSX ETY PORT or the Ethernet link of the TSX P57 5634 is as follows.



How to Modify the Table

The procedure is as follows:

Step	Action
1	Enter the entry number to be modified in the field provided.
2	Enter the new IP address to be modified in the field provided.
3	Enter the new MAC address to be modified in the field marked: **New Ethernet Address**.
4	Is the server locked in run mode? ● If so: enter the password associated to the address server then go to step 5. ● If not: go to step 5.
5	Confirm the modification with the **Change Entry** button.
6	Click on the **Refresh Address Server Database Table** button to display the modification on the screen.

Recognition of Modifications

Configuration modifications are recognized either after a cold restart of the PLC, or on the next loading of the PLC application.

Common Quantum and Premium Setup Page

5

Common Quantum and Premium Setup Page

Overview

This chapter describes setup pages that are common to both the Quantum and Premium.

What's in this Chapter?

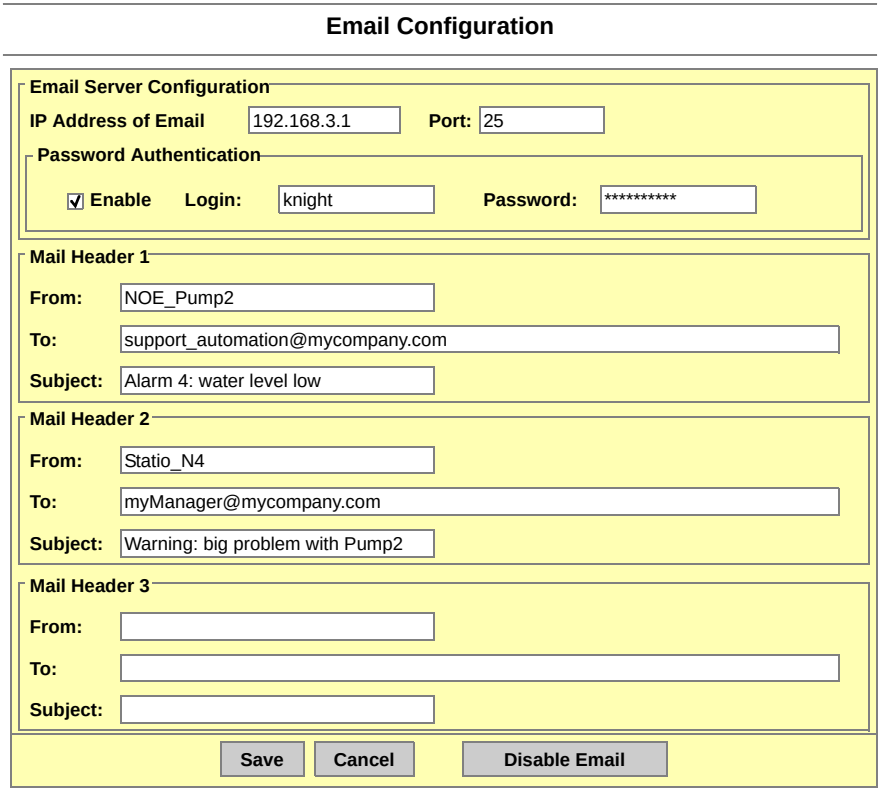
This chapter contains the following topics:

Topic	Page
Configuring the Electronic Mail Notification Service	114
Configuring the Time Synchronization Service	116

Configuring the Electronic Mail Notification Service

Configuring the Mail Service with the Email Configuration Page

You must use the module's embedded Web page to configure the electronic mail notification service. No other method is available.



Mail Service Command Buttons

Mail service configuration buttons:

Button	Description
Save	Saves the new Email configuration. **Note:** Previous configuration is not valid. Previous configuration is not stored.
Cancel	Cancels the entries in the fields. Previous configuration is valid.
Disable Email	Clears the stored configuration, and disables the email service. **Note:** The next time the service is enabled, a new configuration is required.

Configurable Mail Service Parameters

Configure the following parameters:

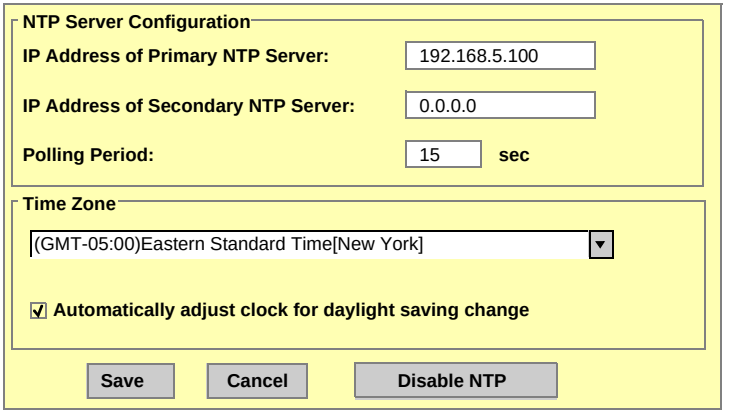
Parameter	Description
IP address of Email	Enter a valid IP address. (This parameter identifies SMTP server.)
Port	Default = 25 (if needed, enter a new value)
Password Authentication	If security is needed, enable Password Authentication. To enable, enter a check mark in the box. Enter values for: ● Login: ● Any printable character allowed, ● 12 character maximum, ● Password: ● Any printable character allowed, ● 12 character maximum.
3 mail headers	Each header must contain **1.** sender's ID in the "From:" field ● 32 character maximum; no spaces, **2.** list of recipients in the "To:" field ● Separate each email address with a comma, ● 128 character maximum, **3.** fixed part of message in the "Subject:" field[1]. ● (32 character maximum).
1. Subject field consists of two parts: **1.** Fixed (32 character maximum) **2.** Dynamic (206 character maximum)	

Configuring the Time Synchronization Service

Configuring the Time Service with the NTP Configuration Page

You must use the module's embedded Web page to configure the time service. No other method is available.

NTP Configuration



Time Service Command Buttons

Execute the following commands

Command Button	Description
Save	Stores new NTP (time service) configuration. Previous configuration is no longer valid.
Cancel	Cancels new NTP (time service) configuration. Previous configuration is valid.
Disable NTP	IP of Primary and Standby set = 0. NTP server not polled Time in controller not updated.

Configurable Time Service Parameters

Configure or change the following parameters on the NTP Configuration page.

1. IP address of primary NTP server
   - Enter a valid IP address,
2. IP address of secondary NTP server
   - Enter a valid IP address,
3. Polling Period (in seconds)
   Enter a value
   - min = 1sec,
   - max = 120sec,
   - default = 5 sec,
4. Time Zone
   - Select from drop-down menu
     Universal Time, Coordinated (GMT) = default,
   - Custom time zone,
5. Automatically adjust clock for daylight saving change
   - Parameter is selected by default (check mark appears) if daylight saving time is chosen.

Changing Time Service Parameters

To make any changes to the time synchronization service,

Step	Action
1	Enter changes in the appropriate field on the NTP Configuration page for one or all of the configurable parameters.
2	Click Save.

Important Information about the Time Service

Note: About the Time Service

1. Enable/Disable Daylight Savings Time parameter
   If the Enable/Disable check box is selected, the module automatically corrects the local time to account for daylight savings time. Therefore, requiring no action, as the daylight savings time start and end are automatically changed each year.
2. Polling Time Parameter
   The time (in seconds) is the time between time updates from the NTP server. The default is 5 Seconds.
3. Storing the Time Service Configuration
   The last time service configuration is saved internally in the Ethernet module.
4. Replacing the Ethernet Module
   If the Ethernet module has to be replaced, the stored configuration is lost, and the system returns to the default configuration.

Customizing Time Zone Parameters

If you want a time zone not listed in the time zone table,

Step	Action	Comment
1	Write the text rules for the custom time zone.	
2	Using an FTP client, store your rules in the file: /FLASH0/wwwroot/conf/NTP/customrules user ID: ntpupdate password: ntpupdate	Root directory to store 'customrules' is set by the FTP server as /FLASH0/wwwroot/conf/NTP
3	When the rules are written, choose the drop down menu on the NTP Configuration web page, and configure (or reboot) the module by selecting Time Zone = Custom	The NTP component looks for customrules, calls the tz compiler and generates a new file called 'tz_custom'. This file is binary file and should not be edited. If the tz compiler detects a syntax error in customrules, the error is logged in the file: /FLASH0/wwwroot/conf/NTP/error.log **1.** NTP component is not launched **2.** NTP Status field in diagnostic web page displays NOT OK.
4	If you want more information, the syntax to write those rules along with a few examples are found in the module in /FLASH0/wwwroot/conf/NTP/instructions.txt	

Time Zone Parameters

Select one from drop-down menu.

Time Zone	Description	DST Available
Custom		Yes
(GMT-12:00)	Dateline Standard Time [Eniwetok Kwajalein]	No
(GMT-11:00)	Samoa Standard Time [Midway Is Samoa]	No
(GMT-10:00)	Hawaiian Standard Time [Hawaii Honolulu]	No
(GMT-09:00)	Alaskan Standard Time [Anchorage]	Yes
(GMT-08:00)	Pacific Standard Time [Los Angeles Tijuana]	Yes
(GMT-07:00)	Mexican Standard Time [Chihuahua La Paz Mazatlan]	Yes
(GMT-07:00)	Mountain Standard Time [Arizona Phoenix]	No
(GMT-07:00)	Mountain Standard Time [Denver]	Yes
(GMT-06:00)	Central Standard Time [Chicago]	Yes
(GMT-06:00)	Mexico Standard Time [Tegucigalpa]	No
(GMT-06:00)	Canada Central Standard Time [Saskatchewan Regina]	No
(GMT-06:00)	Central America Standard Time [Mexico_city]	Yes
(GMT-05:00)	SA Pacific Standard Time [Bogota Lima Quito]	No
(GMT-05:00)	Eastern Standard Time [New York]	Yes
(GMT-05:00)	Eastern Standard Time [Indiana (East)] [Indianapolis]	No
(GMT-04:00)	SA Western Standard Time [Caracas La Paz]	No
(GMT-04:00)	Pacific SA Standard Time [Santiago]	Yes
(GMT-03:30)	Newfoundland Standard Time [Newfoundland St Johns]	Yes
(GMT-03:00)	E. South America Standard Time [Brasilia Sao_Paulo]	Yes
(GMT-03:00)	SA Eastern Standard Time [Buenos Aires Georgetown]	No
(GMT-02:00)	Mid-Atlantic Standard Time [South_Georgia]	No
(GMT-01:00)	Azores Standard Time [Azores Cape Verde Island]	Yes
(GMT)	Universal Coordinated Time [Casablanca, Monrovia]	No
(GMT0)	Greenwich Mean Time [Dublin Edinburgh Lisbon London]	Yes
(GMT+01:00)	Romance Standard Time [Amsterdam CopenHagen Madrid Paris Vilnius]	Yes
(GMT+01:00)	Central European Standard Time [Belgrade Sarajevo Skopje Sofija Zagreb]	Yes
(GMT+01:00)	Central Europe Standard Time [Bratislava Budapest Ljubljana Prague Warsaw]	Yes
(GMT+01:00)	W. Europe Standard Time [Brussels Berlin Bern Rome Stockholm Vienna]	Yes

Time Zone	Description	DST Available
(GMT+02:00)	GTB Standard Time [Athens Istanbul Minsk]	Yes
(GMT+02:00)	E. Europe Standard Time [Bucharest]	Yes
(GMT+02:00)	Egypt Standard Time [Cairo]	Yes
(GMT+02:00)	South Africa Standard Time [Johannesburg Harare Pretoria]	No
(GMT+02:00)	FLE Standard Time [Helsinki Riga Tallinn]	Yes
(GMT+02:00)	Israel Standard Time [Israel Jerusalem]	Yes
(GMT+03:00)	Arabic Standard Time [Baghdad]	Yes
(GMT+03:00)	Arab Standard Time [Kuwait Riyadh]	No
(GMT+03:00)	Russian Standard Time [Moscow St. Petersburg Volgograd]	Yes
(GMT+03:00)	E. Africa Standard Time [Nairobi]	No
(GMT+03:30)	Iran Standard Time [Tehran]	Yes
(GMT+04:00)	Arabian Standard Time [Abu Dhabi Muscat]	No
(GMT+04:00)	Caucasus Standard Time [Baku Tbilisi]	Yes
(GMT+04:00)	Afghanistan Standard Time [Kabul]	No
(GMT+05:00)	Ekaterinburg Standard Time [Ekaterinburg]	Yes
(GMT+05:00)	West Asia Standard Time [Islamabad Karachi Tashkent]	No
(GMT+05:30)	India Standard Time [Bombay Calcutta Madras New Delhi]	No
(GMT+06:00)	Central Asia Standard Time [Almaty Dhaka]	Yes
(GMT+06:00)	Sri Lanka Standard Time [Columbo]	No
(GMT+07:00)	SE Asia Standard Time [Bangkok Hanoi Jakarta]	No
(GMT+08:00)	China Standard Time [Beijing Chongqing Hong Kong Urumqi]	No
(GMT+08:00)	W. Australia Standard Time [Perth]	No
(GMT+08:00)	Singapore Standard Time [Singapore]	No
(GMT+08:00)	Taipei Standard Time [Taipei]	No
(GMT+09:00)	Tokyo Standard Time [Osako Sapporo Tokyo]	No
(GMT+09:00)	Korea Standard Time [Seoul]	No
(GMT+09:00)	Yakutsk Standard Time [Yakutsk]	Yes
(GMT+09:30)	Cen. Australia Standard Time [Adelaide]	Yes
(GMT+09:30)	AUS Central Standard Time [Darwin]	No
(GMT+10:00)	E. Australia Standard Time [Brisbane]	No
(GMT+10:00)	AUS Eastern Standard Time [Canberra Melbourne Sydney]	Yes
(GMT+10:00)	West Pacific Standard Time [Guam Port Moresby]	No
(GMT+10:00)	Tasmania Standard Time [Hobart]	Yes
(GMT+10:00)	Vladivostok Standard Time [Vladivostok]	Yes

Time Zone	Description	DST Available
(GMT+11:00)	Central Pacific Standard Time [Magadan Solomon Is New Caledonia]	Yes
(GMT+12:00)	New Zealand Standard Time [Auckland Wellington]	Yes
(GMT+12:00)	Fiji Standard Time [Fiji Kamchatka Marshall Is]	No

Default Web Site for Micro

6

Default Web Site for Micro

Overview

When you receive the device, it already contains a default Web site with the on-line configuration and diagnostics pages pre-loaded.

You may view these pages simply by installing the module and configuring its IP address. To access the site, type the IP address of the module in your browser and enter the default user name and password of "USER". However, Schneider Electric recommends that you complete the setup procedures as outlined in *Creating a New Configuration, p. 147*.

Note: This section describes only the Web pages common for all FactoryCast platforms : Rackviewer, Data Editor and Graphic Editor. Other pages are described in TSX ETZ User's Guide.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
6.1	Micro Home Page	124
6.2	Monitoring Page	126
6.3	Diagnostics Page	128
6.4	Setup Page	142

6.1 Micro Home Page

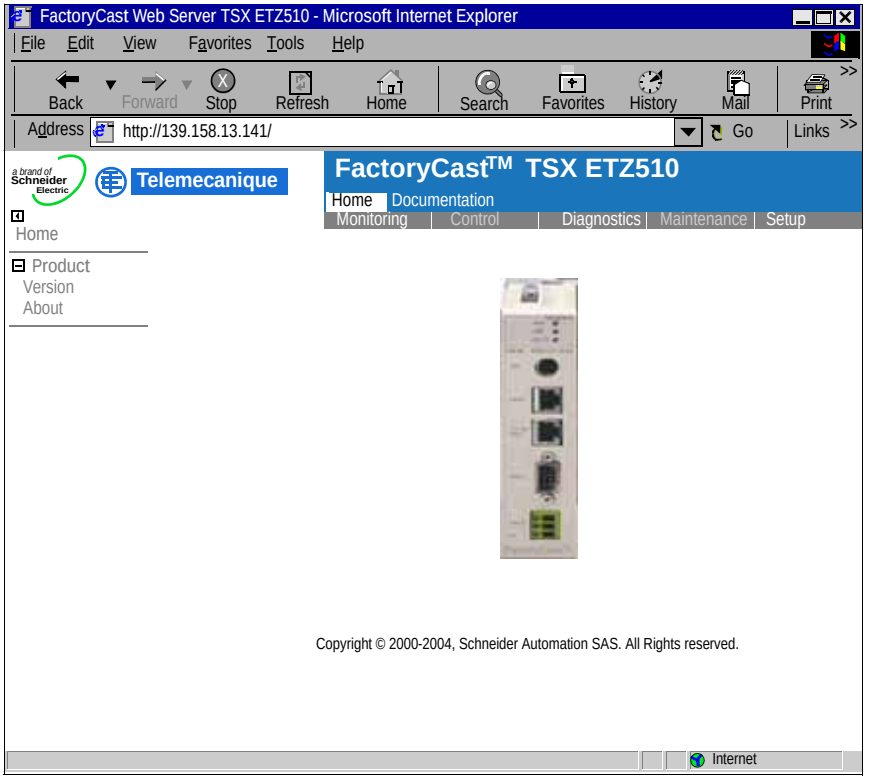
Micro Home Page

Overview

The visitor accesses the Micro home page by entering the IP address of the module in his web browser. No password is required to display this page.

Home Page

This Micro home page looks like this.



Links

From the Micro home page, you can access the following pages :

- ‘Monitoring’,
- ‘Diagnostics’,
- ‘Setup’,
- ‘Documentation’.

The visitor will have to supply a user name and a password to access the services proposed in these pages.

6.2 Monitoring Page

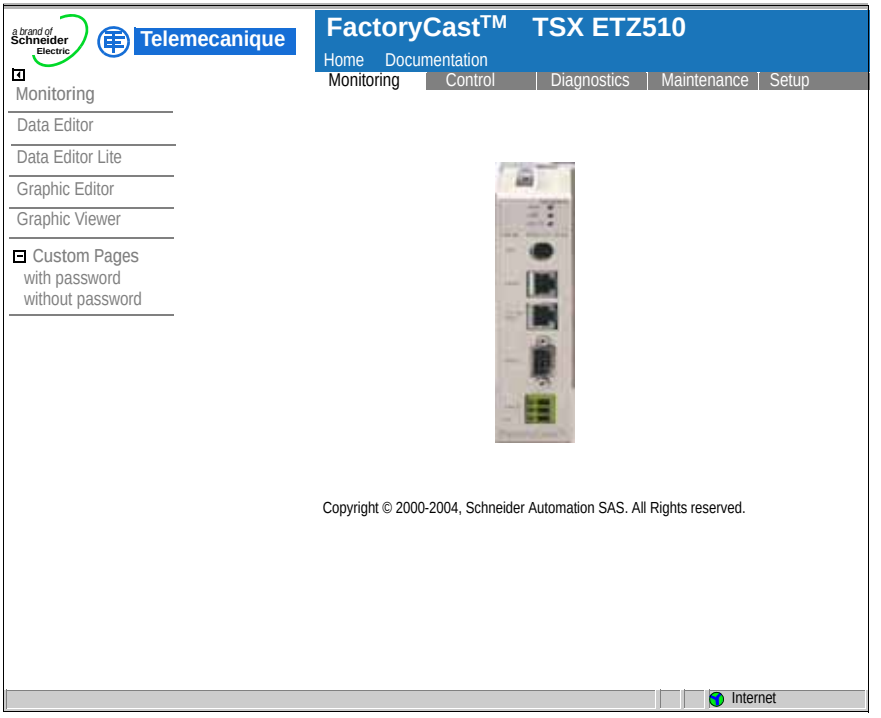
Monitoring Home Page

Home Page

This page lists the various viewer services supported by the default web site of the module and provides links for accessing the services you require.

Illustration

The ‘Monitoring’ home page looks like this:



To access the service you require, click on a link.

- Data editor (See *Data Editor, p. 192*) : for creating variable data tables, so as to be able to determine their value when the table is animated.
- Data editor lite (See *Data Editor Lite, p. 209*) : for creating variable data tables, so as to be able to determine their value when the table is animated (lighter than Data editor).
- Graphic editor (See *Graphic Editor, p. 220*) : for creating graphics, so as to be able to determine the value of variables when the graphic is animated.
- Graphic viewer (See *Graphic Viewer, p. 268*) : for viewing graphics, so as to be able to determine the value of variables when the graphic is animated.
- Password-protected custom pages (See *Adding Custom Pages to the Site, p. 271*) : for viewing screen pages (accessible with password) created by the user.
- Custom pages without password protection (See *Adding Custom Pages to the Site, p. 271*) : for viewing screen pages (accessible to everyone) created by the user.

6.3 Diagnostics Page

Overwiew of the Diagnostics Page

Overview

This section describes the different services proposed by the ‘Diagnostics’ page.

What's in this Section?

This section contains the following topics:

Topic	Page
Diagnostics Home Page	129
Rack Viewer Page	131
PLC Personality Page	132
PLC Communication Pages	133
PLC Analog I/O Page	135
PLC Counter Page	136
Rack Viewer	137
Digital I/O Module Diagnostics Page	139
Analog I/O Module Diagnostics Page	140
Standard Module Diagnostics Page	141

Diagnostics Home Page

Home Page

This page lists the various services supported by the default web site of the module and provides links for accessing the services you require.

Illustration

The ‘Diagnostics’ home page looks like this :



Links

To access the service you require, click on a link.

- ‘Ethernet Statistics’,
- ‘Unitelway Statistics’,
- ‘PPP/Modem Log File’,
- ‘Rack viewer’,
- ‘FDR Statistics’.
- ‘MIB Upload’.

Note: The ‘Rack viewer’ service is the only described. Other services are in TSX ETZ User’s Guide.

Rack Viewer Page

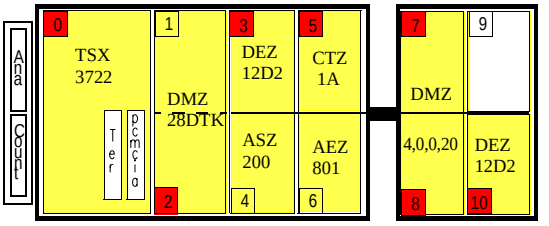
Overview

The Rack Viewer Page displays the current configuration of the TSX Micro.

Sample Page

Here is an example of a Rack Viewer page.

RACK VIEWER



Data

The following information is provided for each module displayed in the rack.

- A vertical label displays the module type and part number.
- The box in the upper left hand corner of the module displays the slot number and module health:
  - a yellow box indicates that the module is functioning properly,
  - a red box indicates that the module is not functioning properly.

Links

Click a module icon to obtain detailed information about that module.

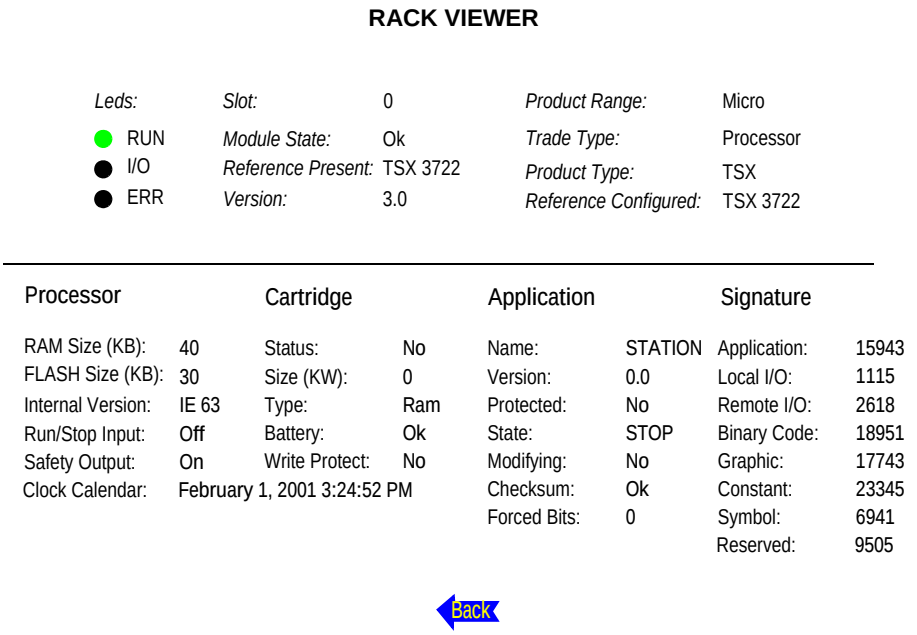
PLC Personality Page

Overview

The PLC Personality page provides information about the controller and its configuration.

Sample Page

Here is an example of a PLC Personality page.



Dynamic Data

The LEDs in the upper left hand corner of the screen provide a dynamic report on the controller status.

LEDs	Color if On	Meaning if On	Meaning if Blinking	Meaning if Off
RUN	Green	Application running	Stopped	PLC error
I/O	Red	I/O event	---	No error
ERR	Red	PLC error	Not configured	No error

Links

The back arrow will take you to the Rack Viewer page for this controller.

PLC Communication Pages

Overview

The TER and PCMCIA pages provide information about the communication links and their configuration.

TER Page

Here is an example of a TER page.

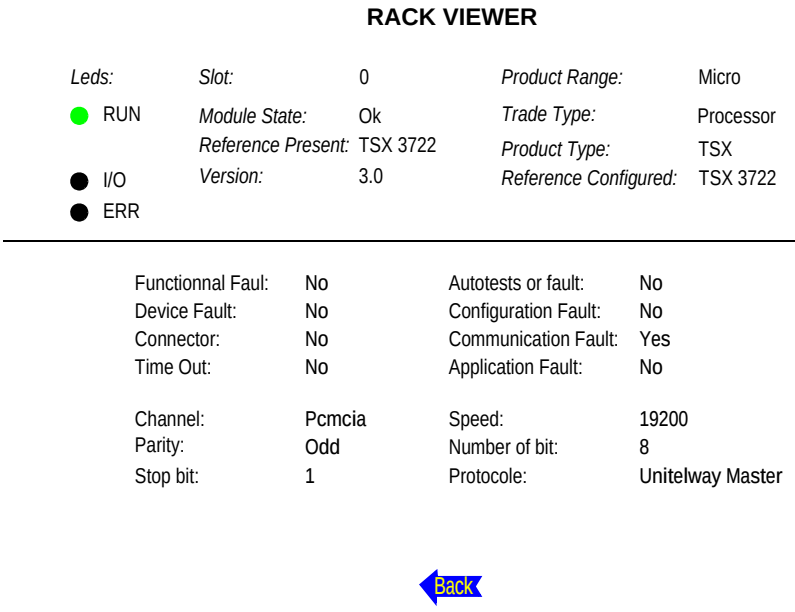
RACK VIEWER

Leds:	*Slot:*	0	*Product Range:*	Micro
RUN	*Module State:*	Ok	*Trade Type:*	Processor
I/O	*Reference Present:*	TSX 3722	*Product Type:*	TSX
ERR	*Version:*	3.0	*Reference Configured:*	TSX 3722

Functionnal Faul:	No	Autotests or fault:	No
Device Fault:	No	Configuration Fault:	No
Connector:	No	Communication Fault:	No
Time Out:	No	Application Fault:	No
Channel:	Ter	Speed:	9600
Parity:	Odd	Number of bit:	8
Stop bit:	1	Protocole:	Unitelway Master



PCMCIA Page Here is an example of a PCMCIA page.



Dynamic Data The LEDs in the upper left hand corner of the screen provide a dynamic report on the controller status.

LEDs	Color if On	Meaning if On	Meaning if Blinking	Meaning if Off
RUN	Green	Application running	Stopped	PLC error
I/O	Red	I/O event	---	No error
ERR	Red	PLC error	Not configured	No error

Links The back arrow will take you to the Rack Viewer page for this controller.

PLC Analog I/O Page

Overview

The Analog I/O page provide information about the embedded Analog I/O and its configuration.

Analog I/O Page

Here is an example of a Analog I/O page.

RACK VIEWER

Leds:	*Slot:*	0	*Product Range:*	Micro
RUN	*Module State:*	Ok	*Trade Type:*	Processor
I/O	*Reference Present:*	TSX 3722	*Product Type:*	TSX
ERR	*Version:*	3.0	*Reference Configured:*	TSX 3722

Functionnal Fault:	No	Autotests:	No
Device Fault:	No	Configuration Fault:	No
Connector:	No	Absent:	No

Analog Channel Viewer

Channels:	*Channel 0*	*Channel 1*	*Channel 2*	*Channel 3*	*Channel 4*	*Channel 5*	*Channel 6*	*Channel 7*
%IW	80	40	40	40	40	40	40	40
States:	-	-	-	-	-	-	-	-
%QW	0							

E= Error F= Forced



Dynamic Data

The LEDs in the upper left hand corner of the screen provide a dynamic report on the controller status.

LEDs	Color if On	Meaning if On	Meaning if Blinking	Meaning if Off
RUN	Green	Application running	Stopped	PLC error
I/O	Red	I/O event	---	No error
ERR	Red	PLC error	Not configured	No error

Links

The back arrow will take you to the Rack Viewer page for this controller.

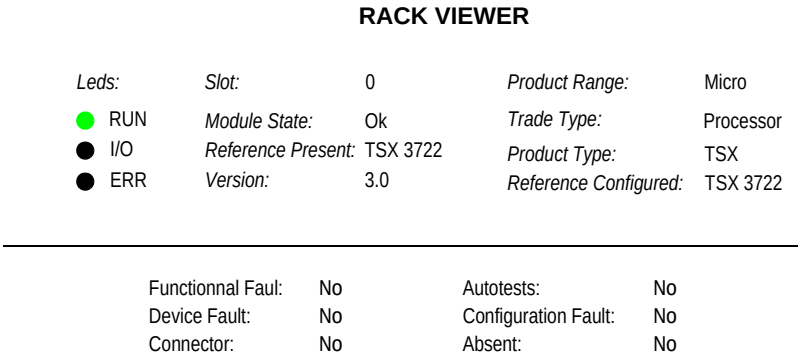
PLC Counter Page

Overview

The Counter page provide information about the embedded counter and its configuration.

Counter Page

Here is an example of a counter page.



Dynamic Data

The LEDs in the upper left hand corner of the screen provide a dynamic report on the controller status.

LEDs	Color if On	Meaning if On	Meaning if Blinking	Meaning if Off
RUN	Green	Application running	Stopped	PLC error
I/O	Red	I/O event	---	No error
ERR	Red	PLC error	Not configured	No error

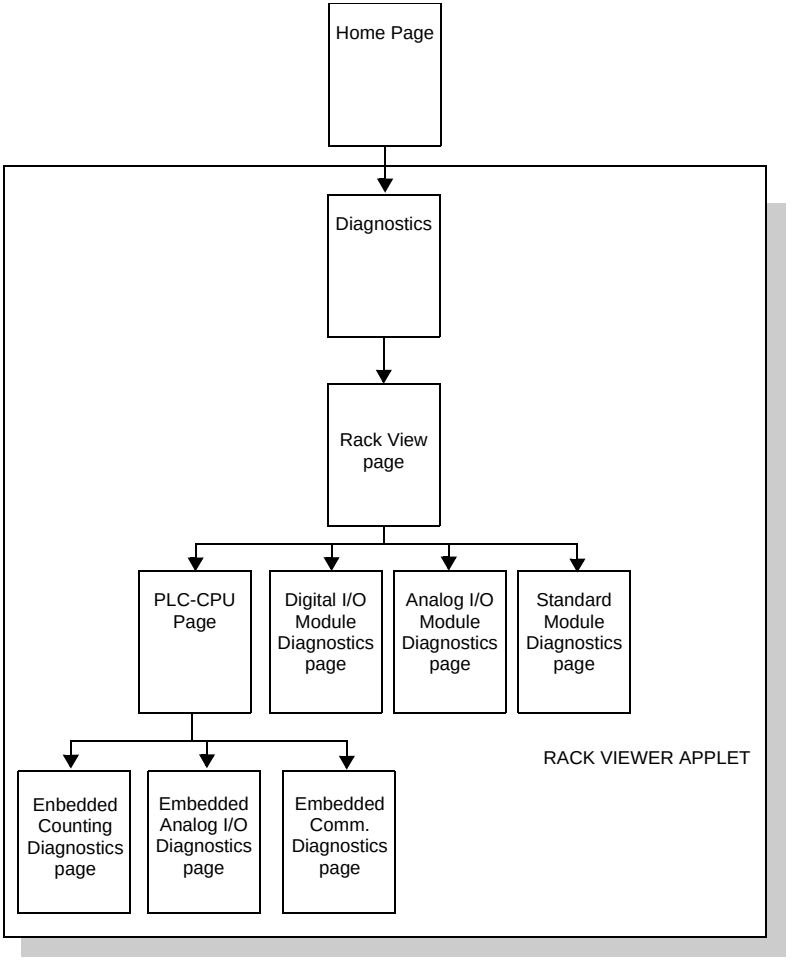
Links

The back arrow will take you to the Rack Viewer page for this controller.

Rack Viewer

Rack Viewer Navigation

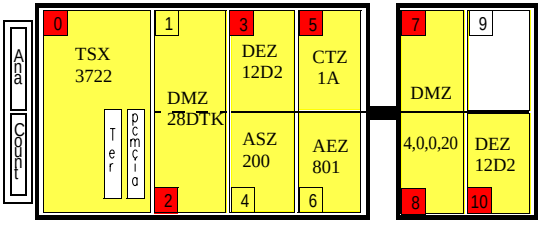
The following illustration represents the navigation page tree of the Rack Viewer Applet.



Rack Viewer

Click the ‘Rack viewer’ link. The applet starts and displays the current configuration of the TSX Micro.

RACK VIEWER



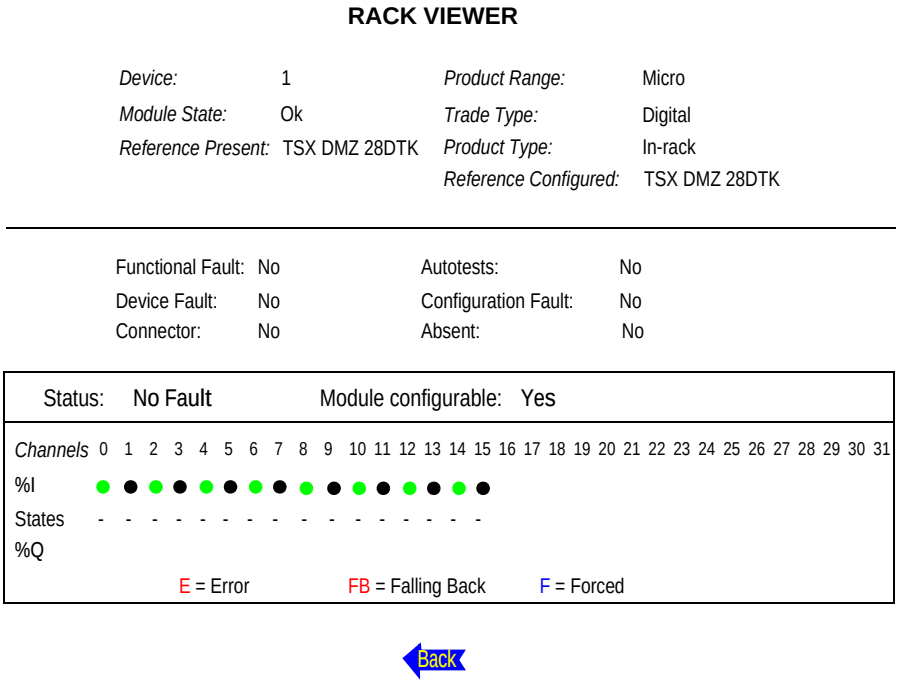
Digital I/O Module Diagnostics Page

Overview

If you click a digital I/O module in the Rack Viewer, you will reach a Digital I/O Module Diagnostics page with detailed information about that module.

Sample Page

Here is an example of Digital I/O Module Diagnostics page.



Links

The back arrow will take you to the Rack Viewer page for this controller.

Analog I/O Module Diagnostics Page

Overview

If you click an analog I/O moduleicon in the Rack Viewer, you will reach an analog I/O Module Diagnostics page with detailed information about that module.

Sample Page

Here is an example of Analog I/O Module Diagnostics page.

RACK VIEWER

Device:	6	*Product Range:*	Micro
Module State:	Ok	*Trade Type:*	Analog
Reference Present:	TSX AEZ 801	*Product Type:*	In-rack
		Reference Configured:	TSX AEZ 801

Functionnal Fault:	No	Autotests:	No
Device Fault:	No	Configuration Fault:	No
Connector:	No	Absent:	No

Analog Channel Viewer

Channels:	*Channel 0*	*Channel 1*	*Channel 2*	*Channel 3*	*Channel 4*	*Channel 5*	*Channel 6*	*Channel 7*
%IW	0	0	0	0	0	0	0	0
States:	-	-	-	-	-	-	-	-
%QW								

E= Error F= Forced



Links

The back arrow will take you to the Rack Viewer page for this controller.

Standard Module Diagnostics Page

Overview

If you click any other type of module in the Rack Viewer page, you reach a standard Module Diagnostic page with detailled information about that module.

Sample Page

Here is an example of a Standard Module Diagnostics page.

RACK VIEWER

Device:	5	*Product Range:*	Micro
Module State:	Ok	*Trade Type:*	Counting
Reference Present:	TSX CTZ 1A	*Product Type:*	In-rack
		Reference Configured:	TSX CTZ 1A

Functionnal Faul:	No	Autotests:	No
Device Fault:	No	Configuration Fault:	No
Connector:	No	Absent:	No



Links

The back arrow will take you to the Rack Viewer page for this controller.

6.4 Setup Page

Setup Home Page

Home Page

This page lists the various services used to configure the module.

Illustration

The ‘Setup’ home page looks like this :



Links

To access the configuration service you require, click on a link :

- ‘Security’,
- ‘IP Configuration’,
- ‘Unitelway Configuration’,
- ‘Automatic Configuration’,
- ‘SNMP Configuration’,
- ‘Reboot’.

Note: These configuration services are described in TSX ETZ User’s Guide.

FactoryCast Configurator

7

FactoryCast Configurator

Overview

The FactoryCast Configurator gives you the ability to manage your Web site.

This section describes how to:

- set up a Web site,
- create a Web-enabled database with symbols (variables) and direct addresses,
- download data to the Embedded Server,
- maintain the site.

If you only want users to view the default Web pages and to view direct addresses in the Data Editor—if you are not planning to customize the site in any way or to view symbols (variables) or to modify data online—you only need to complete the setup procedures.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
7.1	Creating a New Configuration	146
7.2	Creating a Namespace	158
7.3	Transferring Files to the Web Server	172
7.4	FactoryCast Embedded Server Maintenance	180

7.1 Creating a New Configuration

Overview of Creating a New Configuration

Overview

Follow the procedures in this section to create a new FactoryCast configuration.

What's in this Section?

This section contains the following topics:

Topic	Page
Creating a New Configuration	147
Setting Passwords	150
Setting the IP Address	153
Setting File Locations	154
Setting Default Symbol Access	155
Setting Symbol Access Level	156
Saving the Settings	157

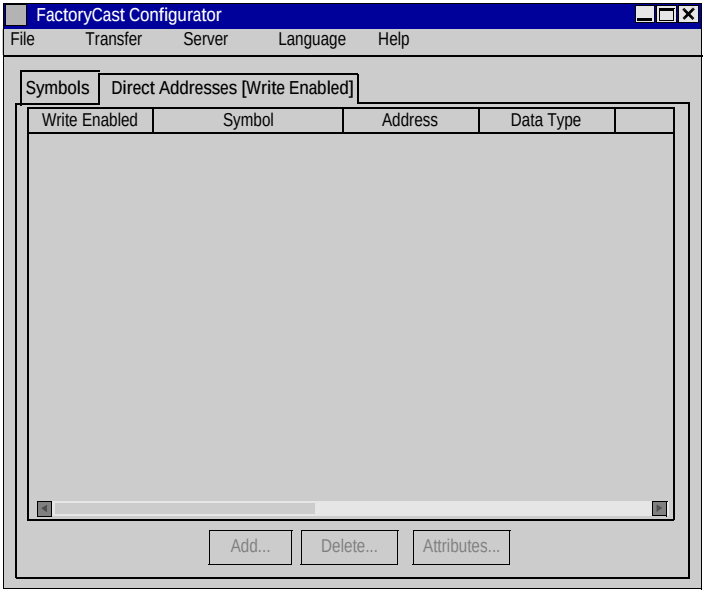
Creating a New Configuration

Overview This section describes how to start the FactoryCast Configurator.

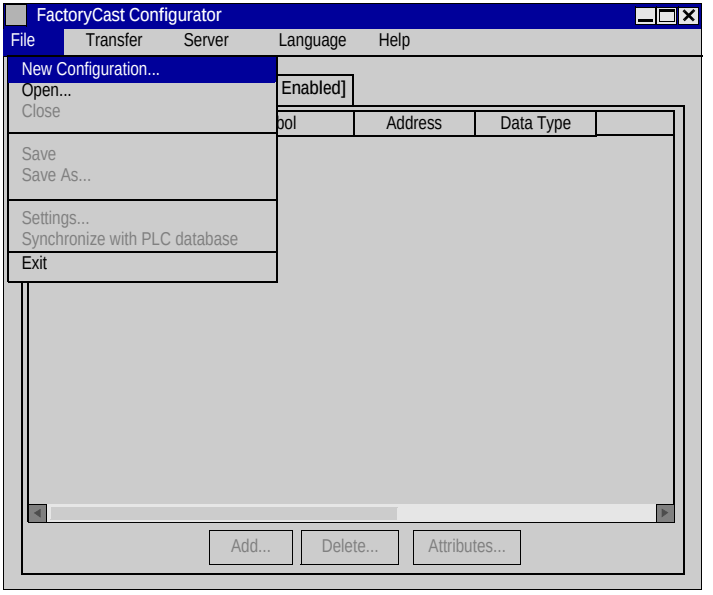
Creating a New Configuration

Select the FactoryCast Configurator application from **Programs** in the **Start** menu.

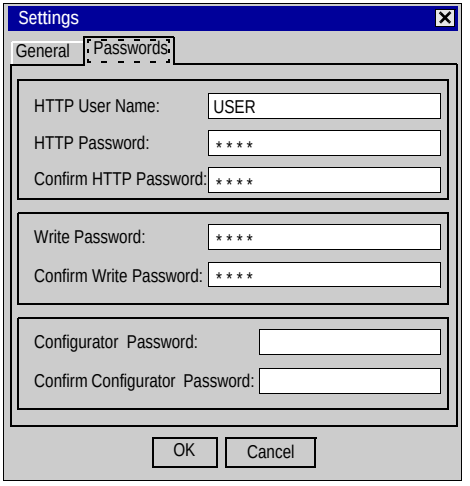
Result: The FactoryCast Configurator main window appears.



From the FactoryCast Configurator menu bar, select **File | New Configuration**.



Result: The Settings dialog box appears and opens to the default tab, which is **Passwords**.



Setting Passwords

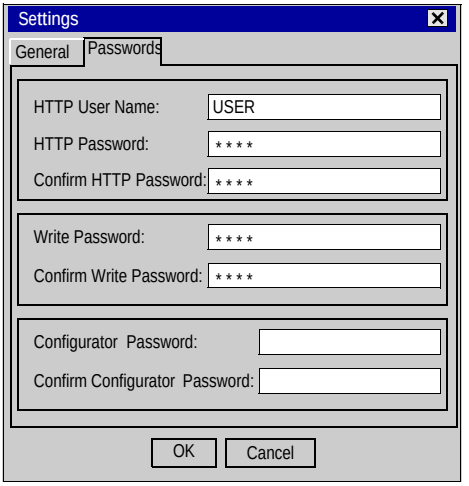
Overview

The first step in setting up your Web site is to specify the passwords for viewing and modifying data.

When you create a new configuration, the Settings window appears automatically with the **Passwords** dialog. Use this dialog to set the passwords.

Passwords Dialog

The following illustration shows the **Settings** dialog box and the fields in the **Passwords** dialog.



Default Settings

Until you apply your own password settings:

- the default **HTTP User Name** is USER,
- the default **HTTP Password** is USER,
- the default **Write Password** is USER,
- there is **no** default **Configurator Password**.

Read Access

Complete the first three fields of the dialog box to set the user name and password for viewing the Web site. Anyone who wants to view the Web site will be prompted for this user name and password.

- **HTTP User Name** is limited to fifteen characters.
- **HTTP Password** is also limited to fifteen characters.
- Confirm the HTTP password by entering it again in the **Confirm HTTP Password** field.

Note: To read Web pages you do not need the Configurator password.

Write Access

Complete the **Write Password** and **Confirm Write Password** in the fourth and fifth fields of the dialog box to set a password for modifying the Web site. Anyone who wants to modify the Web site with the Data Editor or Graphic Editor must supply this password to gain write access.

- **Write Password** is limited to fifteen characters.
- Confirm the write password by entering it again in the **Confirm Write Password** field.

Configurator Access

The Configurator Password, when provided, prompts you to enter the Configurator Password before downloading any of the following features into the Embedded Server.

- Web files,
- Namespace,
- Security,
- Restore,
- Restore Defaults,
- Set XWay Address,
- Plug-ins.

We Recommend

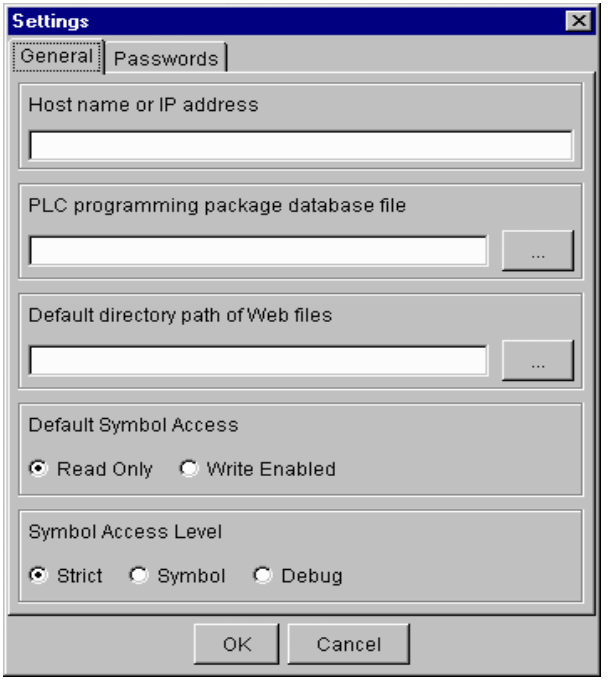
After initially entering the Configurator Password, FactoryCast will remember that password for the duration of the session. It is recommended to close the FactoryCast Configurator after each session to prevent unauthorized people from using your password to gain write access to your Embedded Server.

Accessing the General Dialog

Accessing the General tab allows you to enter any of the following:

- **Host name or IP address**,
- **PLC programming package database file**,
- **Default directory path of Web files**,
- **Default Symbol Access**,
- **Symbol Access Level**.

Select the **General** tab on the **Settings** window to access the General settings dialog.



The following sections of this manual describe setting these fields.

Setting the IP Address

Overview

In the **Settings** dialog box, you will want to specify either the **Host Name** or **IP Address** of the Embedded Server.

Setting the IP Address

Enter the **Host Name** or **IP Address** of the Embedded Server.

Note: The IP address of the Quantum or Premium or Micro Embedded Server and any host name should be assigned by your system administrator. For performance reasons, if you do not have a DNS, you should add the Host Name and IP address to your Hosts file on your Windows PC.

Setting File Locations

Overview

If you want to customize your Web site, use the **Settings | General** dialog to tell the FactoryCast Configurator where to find the following files or directories.

Use the following guidelines.

Give the location of	In order to add
Programming software database file	Symbols (variables) from a Concept, PL7 or Unity Pro database to the Namespace

Use the following guidelines.

Give the location of	In order to add
Directory containing custom Web files	Custom Web files

Specifying File Locations

Enter the directory path in the text entry field, or use the ellipsis buttons to the right of the text boxes to browse for the desired files and directories.

Setting Default Symbol Access

Overview

If you want to view and modify symbols (variables) in your Web site, use the **Settings | General** dialog box to set the Default Symbol Access. This setting determines the initial access rights for all the symbols you will add to the Namespace.

Options

Select one of the following radio buttons.

- **Write Enabled** sets write permission for all symbols (variables) as you add them to the configuration. A user who supplies the write password would be able to modify these symbols.
- **Read Only** sets read-only access for all symbols (variables) as you add them to the configuration. Even if a user supplies the write password, he would not be allowed to modify these symbols.

These settings can be modified later for individual symbols.

Setting Symbol Access Level

Overview

In order to view or modify symbols (variables) in your Web site, it is necessary that the Namespace file that is available on the Embedded Server be consistent with the PLC application program that is currently running. The Symbol Access Level determines the type of access that is permitted under different levels of consistency between the Namespace and the PLC program.

Symbol Access Level

Select one of the following radio buttons:

- Strict requires that there be an exact match between the versions of the Namespace and the PLC program in order to view or modify any symbols (variables).
- Symbol requires only that the memory layout of the symbols (variables) in the Namespace be the same as the PLC program in order to view or modify any symbols (variables). The Namespace can be for a different version of the PLC program, but the addresses of the symbols (variables) must be the same. This setting only has an effect for a Unity Pro PLC, and is the same as Strict for any other type of PLC.
- Debug removes any consistency requirement between Namespace and PLC program in order to view the symbols (variables). If the Namespace is for a different PLC program or the memory layout of the symbols is different, access to the symbols (variables) is restricted to read only. With this setting, if the Namespace and PLC program are not consistent, then it is possible that the value read from PLC for a symbol (variable) is not the actual value. Therefore, this setting should be used with caution, and only for a Web site that is under development.

Saving the Settings

Overview

If you are satisfied with your settings, click **OK** to save them and to exit the **Settings** dialog box.

Planning Ahead

- If you are using the Premium ETY Embedded Server or the Micro ETZ Embedded Server, you must set the XWay Address (see *Setting the XWay Address, p. 186*).
- If you are using the Quantum NOE Embedded Server and you **are** adding a database of symbols (variables) and write-enabled direct addresses to your Web site, you may proceed to *Creating a Namespace, p. 158*.
- If you are using the Quantum NOE Embedded Server and you are **not** adding a database of symbols (variables) or write-enabled direct addresses to your Web site, you may proceed to *Adding Custom Pages to the Site, p. 271*

Note: At this point, save your configuration file to disk (using the **File** → **Save As...**).

7.2 Creating a Namespace

Creating a Namespace

Overview

A namespace is a Web-enabled database containing symbols (variables) and/or write-enabled direct addresses.

This section describes how to use symbols (variables) from a software database and direct addresses to create a namespace for your Web site.

What's in this Section?

This section contains the following topics:

Topic	Page
Importing Symbols (Variables)	159
Setting Symbol (Variable) Access	162
Adding Direct Addresses	163
Direct Address Blocks and Symbol (Variable) Security	167
Editing and Deleting Direct Addresses	169
Synchronizing Your Namespace with Its Database	170

Importing Symbols (Variables)

Overview

If you want to view symbols (variables) in the Data Editor, Graphic Editor, or Alarm Viewer, you must include them in the namespace. The Data Editor will not allow you to view symbols (variables) that are not found in the namespace.

This topic describes how to:

- import symbols into a namespace,
- delete symbols from a namespace.

Accessing Your Software Database

To access a Concept database, the Concept software must be installed on your computer. Concept database files have a .prj file extension.

A PL7 database is accessible from any mapped drive. PL7 database files have a .fef file extension.

A Unity Pro database is accessible if the Unity Pro P-Server is installed on your computer and the database file (.stu file extension) is accessible from any mapped drive. A Unity Pro P-Unit export file (.xvm file extension) can also be used to import Unity Pro symbols (variables).

Note: Only elementary variables are available for import through the Unity Pro P-Unit export file.

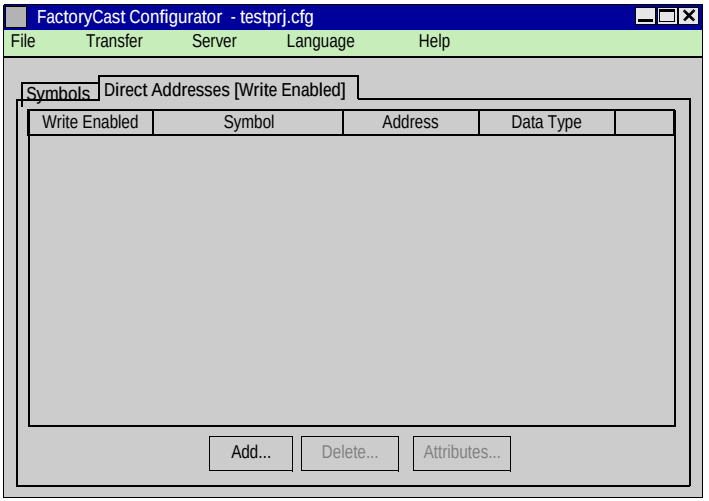
Accessing Your Project Database

In order to access your symbols (variables) from your project database, you must specify the project location in the **File | Settings** dialog box. Select the **General** tab and enter the project database routing path in the "PLC Programming Package Database File" entry field.

Adding Symbols (Variables)

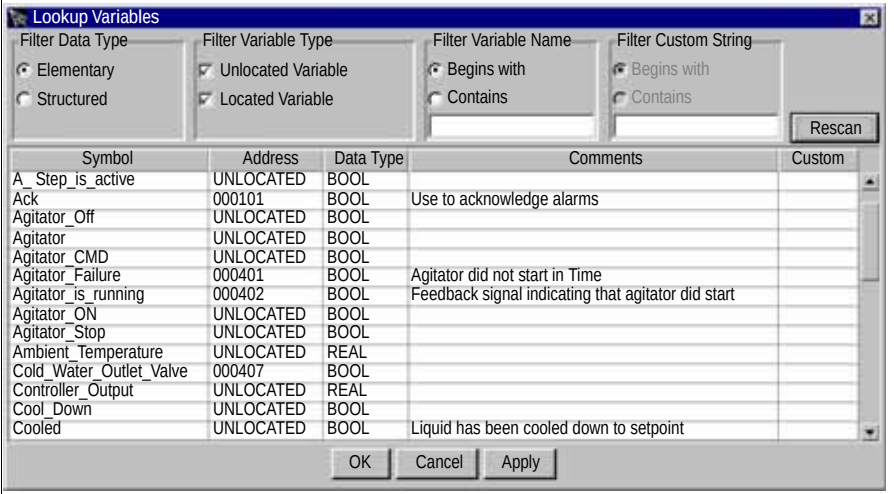
Follow the steps in the table below to import symbols (variables) from a Concept, PL7, or Unity Pro database to your namespace.

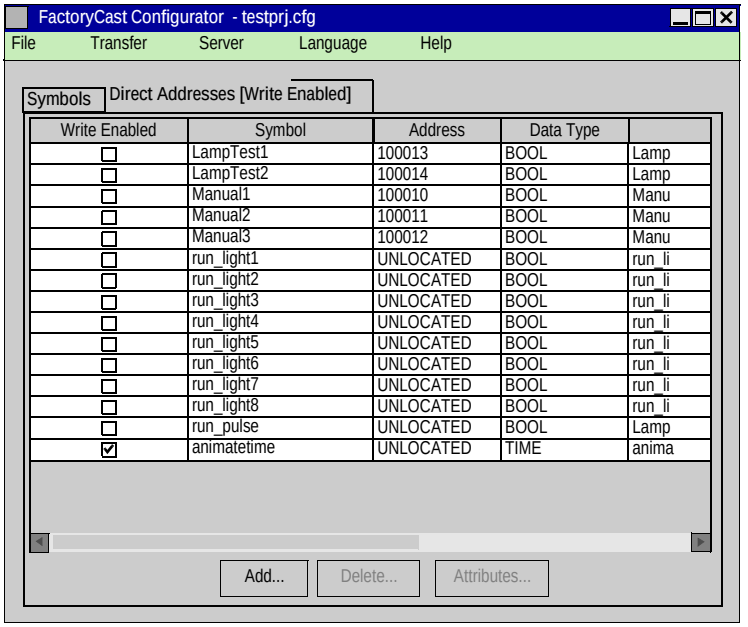
Step	Action
1	Click the **Add** button at the bottom of the **Symbols** window.



Note: The location of the software database file must already be entered in the **File | Settings** window for the **Add** button to be enabled.

Result: The **Lookup Variables** window appears. It displays symbols (variables) associated with the database.



Step	Action
2	Click on the symbols in the **Symbol** column to highlight them individually or in blocks. Press the CTRL key to highlight symbols (variables) that are not adjacent to one another. **Note:** The **Lookup Variables** window also allows the user to filter variable by: • Data Type: Elementary or Structured, • Variable Type: Unlocated or Located, • Variable Name, • Custom String (Unity Pro only). Use the **Rescan** button to apply a filter and rescan the database.
3	Click **Apply** to add the selected symbols (variables) to the namespace without exiting the **Lookup Variables** window. Otherwise, click **OK** to exit the **Lookup Variables** window. **Result:** The symbols (variables) you selected appear in the **Symbols** window. 
4	To add more symbols (variables), click the **Add** button at the bottom of the **Symbols** window and repeat steps 2 and 3.

Deleting Symbols (Variables)

To delete a symbol (variable) from a namespace, select the symbol (variable) and click the **Delete** button at the bottom of the **Symbols** window

Setting Symbol (Variable) Access

Overview

FactoryCast Configurator sets default security for each symbol (variable) as you add it to the namespace, based on your selection for the Default Symbol Access in the **Settings** dialog.

This section discusses security considerations and describes two ways for you to modify the security setting for a symbol.

Two Levels of Access

Each symbol (variable) may be write-enabled or read-only.

- **Write Enabled** means users can modify the symbol value after supplying the write password.
- **Read Only** means users cannot modify symbol (variable) values, even after supplying the write password.

	**CAUTION** **UNAUTHORIZED CHANGES TO SYMBOLS (VARIABLES).** Be careful about which symbols (variables) you allow to be modified online, and be cautious about who has permission to modify them. Unauthorized or incorrect changes to symbols (variables) may change the behavior of your application in ways that may be undesirable or even hazardous. **Failure to follow this instruction can result in injury or equipment damage.**

Using the Attributes Button to Set Access

Follow the steps in the table below to use the Attributes button to set symbol (variable) access.

Step	Action
1	Highlight one or more symbols (variables) in the **Symbol** window.
2	Click the **Attributes** button at the bottom of the window.
3	Click either the **Write Enabled** or **Read Only** button to set the access option.
4	Click **OK**.

Adding Direct Addresses

Overview

If you only want users to view direct addresses, you do not need to include them in the namespace. The Data Editor and Graphic Editor can view any direct address.

If you want users to be able to modify a direct address, you must include it in the namespace. Any direct address included in the namespace is automatically write-enabled.

This section describes how to include direct addresses in a namespace.



CAUTION

UNAUTHORIZED CHANGES TO DIRECT ADDRESSES.

Be careful about which direct addresses you allow to be modified online, and be cautious about who has permission to modify them. Unauthorized or incorrect changes to direct addresses may change the behavior of your application in ways that may be undesirable or even hazardous.

Failure to follow this instruction can result in injury or equipment damage.

Micro Register Values

The Micro register values are shown below.

Address Range	Type
%S0 - %S127	Boolean (BOOL)
%SW0 - %SW127	Word 16 (INT)
%SD0 - %SD126	Word 32 (DINT)
%M0 - %M255	Boolean (BOOL)
%MW0 - %MW17920	Word 16 (INT)
%MD0 - %MD17919	Word 32 (DINT)
%MF0 - %MF17919	Real 32 (REAL)

Premium Register Values

The Premium register values are shown below.

Address Range	Type
%S0 - %S127	Boolean (BOOL)
%SW0 - %SW255	Word 16 (INT)
%SD0 - %SD254	Word 32 (DINT)
%M0 - %M12151	Boolean (BOOL)
%MW0 - %MW31367	Word 16 (INT)
%MD0 - %MD31366	Word 32 (DINT)
%MF0 - %MF31366	Real 32 (REAL)

For a Unity Premium PLC, ranges for %SD, %MD and %MF addresses are not entered. Any "Memory Words" that are write-enabled for a %MW address range are also write-enabled for %MD and %MF addresses, since all %MW,%MD and %MF addresses overlap the same memory area. Likewise, any "System Words" that are write-enabled for a %SW address range are also write-enabled for %SD addresses.

Quantum Register Values

The Quantum (except Unity Quantum) register ranges are shown below

Address Range	Type
000001-065535	Boolean (BOOL)
400001-465535	Word 16 (INT)
600000-699999	Word 16 (INT)

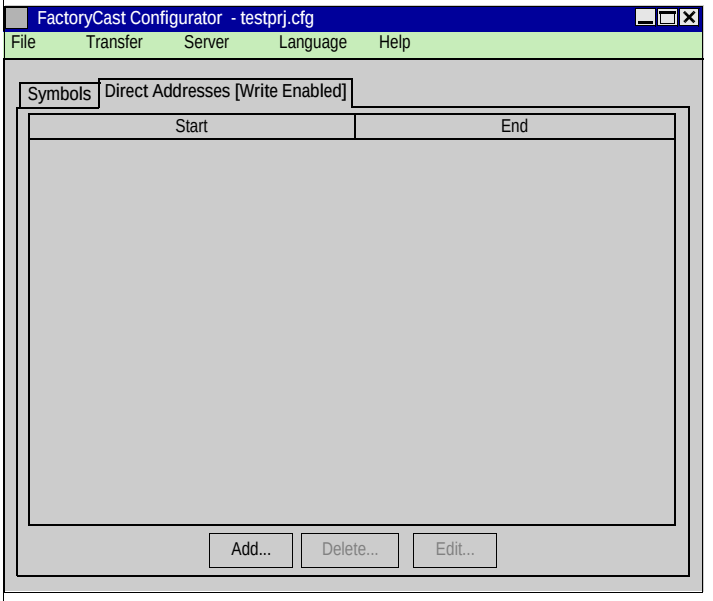
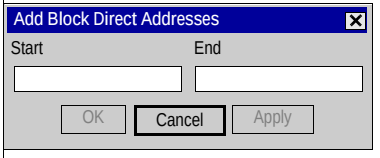
The Unity Quantum register ranges are shown below

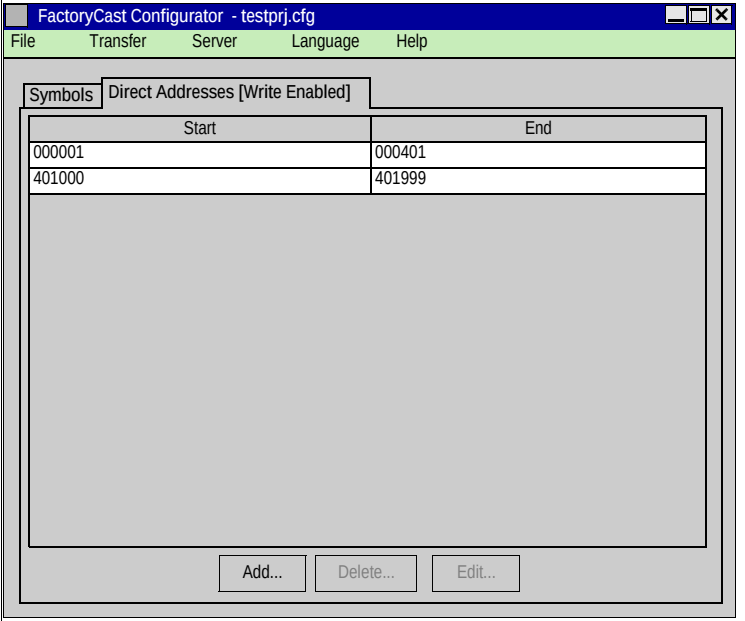
Address Range	Type
%M1-%M65535	Boolean (BOOL)
%MW1-%MW65535	Word 16 (INT)

For Unity Quantum PLC, ranges for %MD and %MF address are not explicitly entered. Any "Memory Words" that are write-enabled for a %MW address range are also write-enabled for %MD and %MF addresses, since all %MW, %MD and %MF addresses overlap the same memory area. Also, a %M address range covers the equivalent Quantum 0x address range, and a %MW address range covers the equivalent Quantum 4x address range. (Unity Quantum does not support 6x addresses.)

Adding Direct Addresses

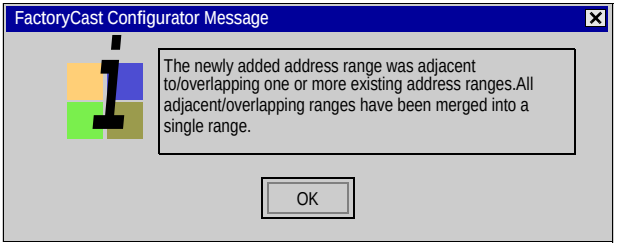
Follow the steps in the table below to import direct addresses into a namespace.

Step	Action
1	Select the **Direct Addresses [Write Enabled]** tab in the FactoryCast Configurator window. **Result:** The **Direct Addresses [Write Enabled]** screen appears. 
2	Click the **Add** button at the bottom of the window. **Result:** The **Add Block Direct Addresses** dialog appears. 
3	Enter a valid range of addresses for your Quantum or Premium or Micro controller. **Note:** If you are only using direct addresses in your namespace, you may specify any valid range for the controller. If you are including symbols (variables) in the namespace, you may only specify direct addresses which are actually configured.

Step	Action
4	Click **Apply** to save the specified registers to the namespace without exiting the dialog. Otherwise, Click **OK** to save the registers and return to the **Direct Addresses [Write Enabled]** window. **Result:** The direct address register ranges are displayed in the window 
5	To add more direct addresses, repeat steps 2 through 4

Overlapping Register Ranges

If the register range that you have specified overlaps with a range already in the namespace, FactoryCast Configurator merges them into a single range and notifies you with the following message. Click **OK** to acknowledge the message.



Direct Address Blocks and Symbol (Variable) Security

Overview

When you import a block of direct addresses into a namespace which already contains symbols (variables), you may inadvertently include addresses associated with symbols (variables). This can create a conflict between the security setting for the symbol (variable) itself and for its address.

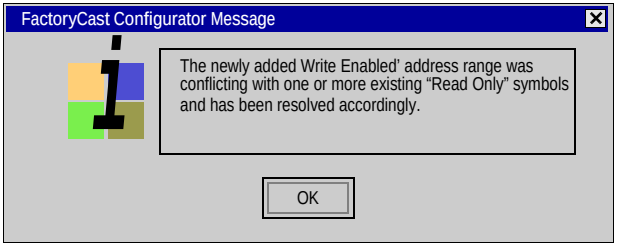
Likewise, when you import symbols (variables) into a namespace which already contains a block of direct addresses, you may include symbols (variables) whose address is included in the direct address range. This also can create a security conflict.

This section describes how the FactoryCast Configurator notifies you and resolves the conflict.

Conflict#1

If your block of direct addresses includes a symbol (variable) which has already been included in the namespace as read-only, FactoryCast Configurator sends you a message and removes the address from the block. The symbol (variable) remains read-only.

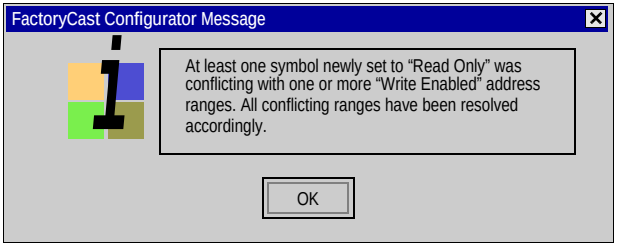
When you receive the message, click **OK**.



Conflict#2

If a symbol (variable) is imported into the namespace as read-only OR if you change its security setting from write-enabled to read-only, and that symbol (variable) is included in a block of direct addresses, FactoryCast Configurator sends you a message and removes the address from the block.

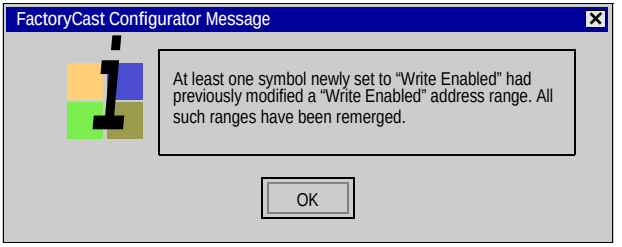
When you receive the message, click **OK**.



Reversing a Conflict

If you change the security setting on a symbol (variable) from read-only to write-enabled and that symbol (variable) had been removed from a block of direct addresses, FactoryCast Configurator sends you a message and will restore it to the block.

When you receive the message, click **OK**.



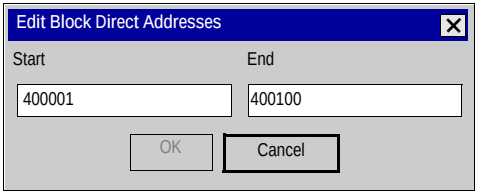
Editing and Deleting Direct Addresses

Overview

This section describes how to edit or delete direct addresses in the namespace.

Editing Direct Addresses

Follow the steps in the table below to edit a block of direct addresses.

Step	Action
1	Select a block of addresses in the Write Enabled Direct Addresses window.
2	Click the **Edit** button at the bottom of the window **Result:** The **Edit Block Direct Addresses** dialog appears 
3	Modify the register range by typing a new value in the **Start** or **End** field.
4	Click **OK**.

Deleting Direct Addresses

To delete a direct address from the namespace, select it in the **Direct Address [Write Enabled]** window and click the **Delete** button at the bottom of the window.

Synchronizing Your Namespace with Its Database

Overview

Over time, you may make changes to the Concept, PL7 or Unity Pro database from which you created your namespace. The FactoryCast Configurator will automatically.

notify you of differences between the database and your namespace when you open a configuration that has a PLC database file set for it.

This section describes how the Configuration tool notifies you and prompts you to synchronize your namespace with the database.

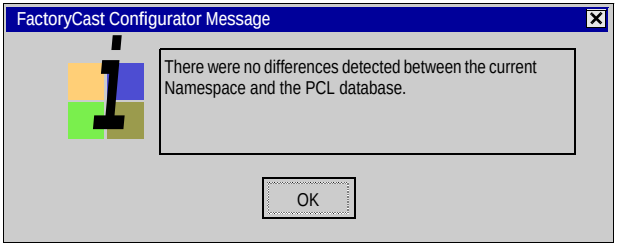
Verifying Discrepancies

Select **File | Synchronize with PLC database** to verify whether differences exist between the current namespace and the corresponding database.

The **Synchronization Status** window appears to notify you if differences are detected between the configuration and the database.

No Differences Detected

If no discrepancy is detected, the following window appears.

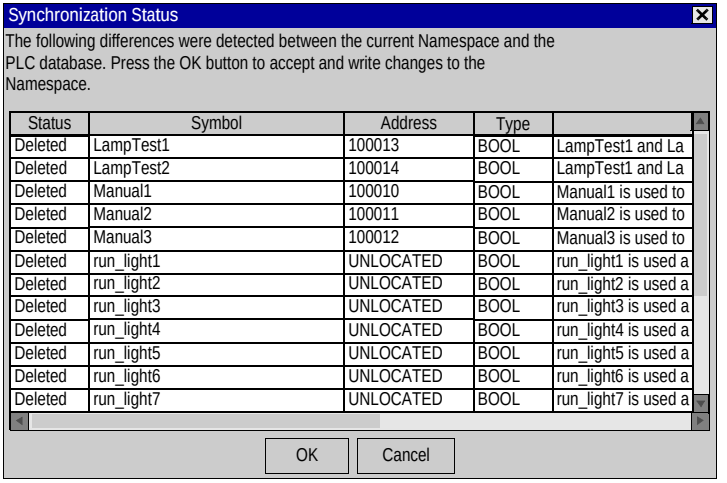


Differences Detected

If a discrepancy between the namespace and the database is detected, a notification appears in the **Synchronization Status** dialog. Discrepancies include

- A symbol in the namespace has been deleted from the database.
- A symbol in the namespace has had its address or data type changed in the database.
- A symbol in the namespace has had its name changed in the database.

The following dialog box displays.



Click the **OK** button to accept the results of synchronization. Synchronizing will alter item(s) that differ from the database.

Note: After synchronizing the namespace with its database, you will want to save the namespace to the configuration file and download the namespace to the Embedded Server.

If You Don't Synchronize

Until the files are synchronized, you will not be able to add symbols (variables) to the namespace.

7.3 Transferring Files to the Web Server

Transferring Files to the Web Server

Overview

This section describes transferring files to the Embedded Server, files such as Custom Web File, loading the Namespace File, and loading the Security File, along with describing processes such as backing up and restoring the Embedded Server's Web directory.

What's in this Section?

This section contains the following topics:

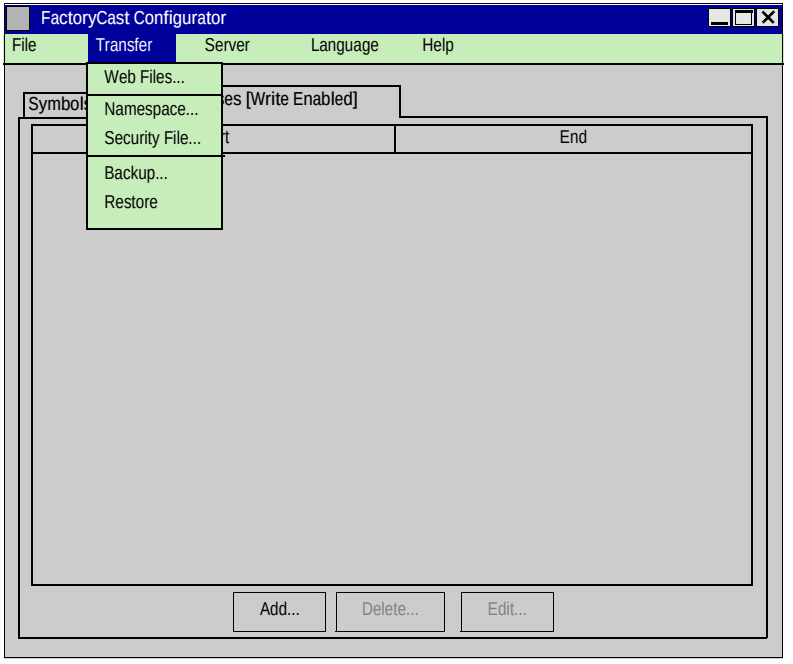
Topic	Page
Custom Web File	173
Loading the Namespace File	176
Loading the Security File	177
Backing Up the Web Server	178
Restoring the Web Server from a Backup	179

Custom Web File

Overview

Transfer menu items enable you to perform functions to control your programs; for example, transferring files to and from the server, and **Backup** and **Restore** of the Web site on the server.

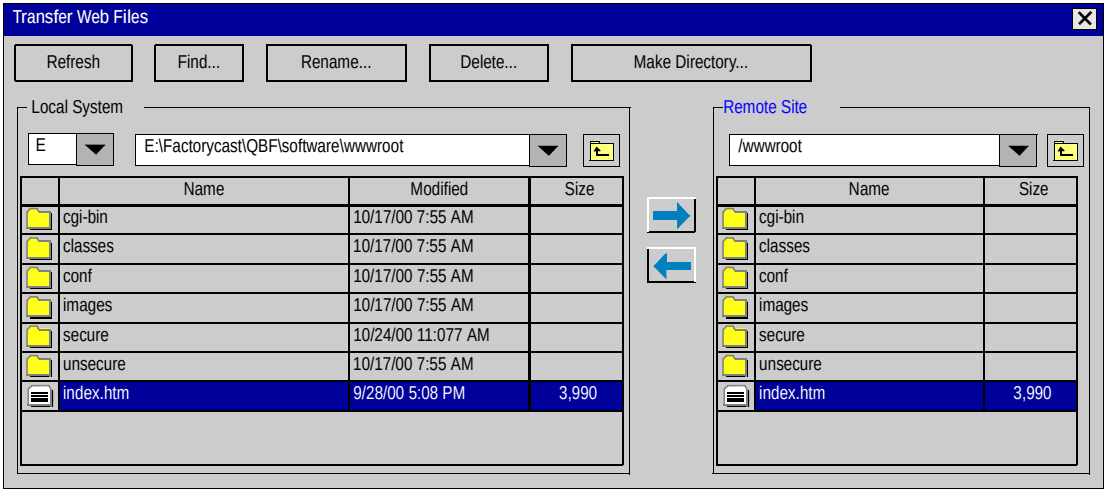
The **Transfer** menu has the following commands.



Transfer Web Files Dialog Box

Transfer Web Files is an FTP utility that allows you to delete, modify, or transfer files to the Embedded Server.

This dialog is displayed when you select the menu item **Transfer | Web Files**:



Main Features of Transfer Web Files

The main features of this dialog box are:

- Two windows provided for listing the contents of a directory:
  - one on the left for the local PC file system
  - one on the right for the server's remote file system
- The local directory is initially set to the **Default Directory of Web Files** specified in the **Settings** dialog. If no configuration is open, or if the specified directory does not exist, then the local directory is set to the FactoryCast Configurator's current working directory.
- The remote directory is initially set to **wwwroot**.
- In addition to the ← and → buttons, standard drag-and-drop is supported to allow you to transfer files between the local and remote file systems.
- Multiple files and directories can be selected for deletion or transfer via a single user operation.
- When a directory is transferred, all of its contents (including subdirectories and their contents, recursively) are transferred.
- When a directory is deleted, all of its contents (including subdirectories and their contents, recursively) are deleted.
- A directory can be created on the local or remote file system by using the **Make Directory...** button.
- A file can be found on the remote file system with a **Find** dialog, which is displayed when the **Find...** button is clicked. This dialog lists the full path names of all files that match a specified search criteria. Standard wildcard searches using the asterisk (*) and question mark (?) characters are supported.

Loading the Namespace File

Overview

After modifying the namespace, you will want to save the namespace as part of the configuration file and then download the namespace to the Embedded Server.

Loading the Namespace File

Follow the steps in the table below to save the namespace file in your configuration file, and then download the namespace file to the Embedded Server.

Step	Action
1	Select **File | Save** from the menu bar. **Result:** The namespace is saved in the configuration file.
2	Select **Transfer | Namespace** to download the namespace to the Embedded Server. **Result:** A Download Confirmation dialog appears with the IP address and any host name of the Embedded Server. **Note:** If the host name or IP address is incorrect, make the appropriate changes in the Settings window. (Refer to *Setting the IP Address, p. 153*)
3	Click **OK** to confirm the IP address and begin the download.

Loading the Security File

Overview

After modifying the security file, you will want to save the security file as part of the configuration file and then download the security file to the Embedded Server.

Loading the Namespace File

Follow the steps in the table below to save the security file in your configuration file, and then download the security file to the Embedded Server.

Step	Action
1	Select **File | Save** from the menu bar. **Result:** The security is saved in the configuration file.
2	Select **Transfer | Security** to download the security file to the Embedded Server. **Result:** A Download Confirmation dialog appears with the IP address and any host name of the Embedded Server. **Note:** If the host name or IP address is incorrect, make the appropriate changes in the Settings window. (Refer to *Setting the IP Address, p. 153*)
3	Click **OK** to confirm the IP address and begin the download.

Backing Up the Web Server

Overview

The **Transfer | Backup** option is used to archive the entire file contents of the Embedded Server. A zip file is created in the directory specified as a result of the backup operation.

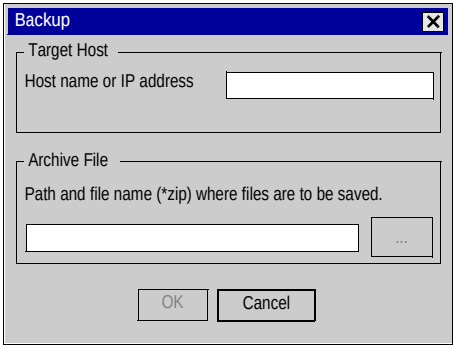
The contents of the backup can be used to restore the Embedded Server to the configuration at the time of the backup.

Do Not Modify Backup Files

You should not make any modifications to the directory structures of files in the zip file.

Create a Backup

Step Action Element

Step	Action
1	Select **Transfer | Backup**. **Result:** The **Backup** dialog box appears. 
2	Enter the **Host Name** or **IP Address** of the Embedded Server.
3	Enter the **Path** and **File Name** where you would like to store the backup file. Use the **Ellipsis** button to the right of the text box to browse for a location.
4	Click **OK** to begin the backup.

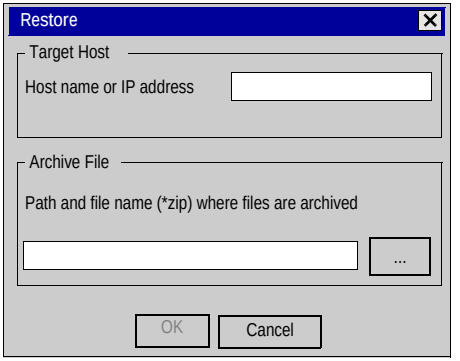
Restoring the Web Server from a Backup

Overview

The **Transfer | Restore** option is used to restore the contents of a backup file to the FactoryCast Configurator.

Restoring the Web Server

Follow the steps below to restore your Web server from a backup.

Step	Action
1	Select **Transfer	Restore**. **Result:** The **Restore** dialog box appears. 
2	Enter the **Host Name** or **IP Address** of the Embedded Server.
3	Enter the **Path** and **File Name** of the backup file. Use the **Ellipsis** button to the right of the text box to browse for the file.
4	Click **OK**.

7.4 FactoryCast Embedded Server Maintenance

Embedded Server Maintenance

Overview

This section describes how to maintain your Embedded Server.

What's in this Section?

This section contains the following topics:

Topic	Page
Embedded Server Operations	181
Restoring Module Defaults	184
Setting the XWay Address	186
Loading the FactoryCast Applet Plug-ins	187

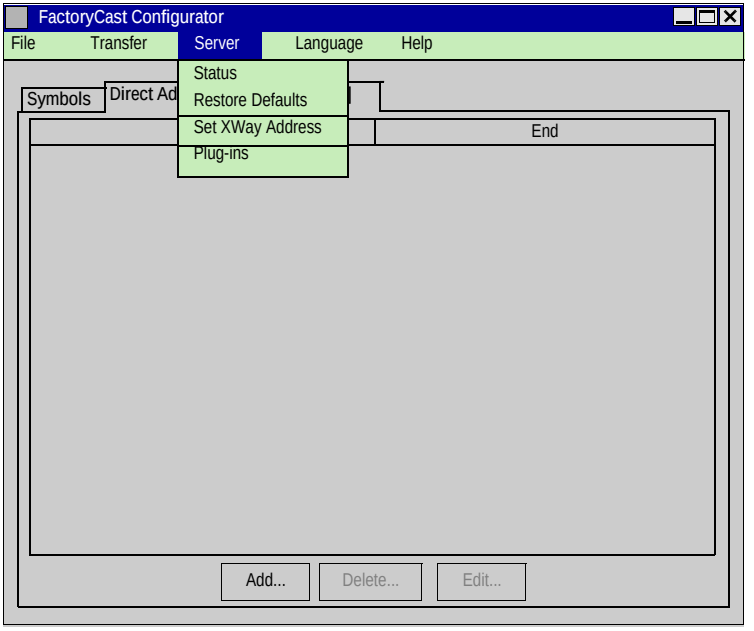
Embedded Server Operations

Overview

The **Server** menu enables you to perform the Embedded Server operations by selecting one of four menu items: **Status**, **Restore** (module) **Defaults**, **Set XWay Address**, and **Plug-ins**. The Plug-ins menu item allows you to choose which options will be downloaded to the Embedded Server.

Embedded Server

The **Server** menu has the following commands.



You can check the following with the FactoryCast Configurator:

- bytes of free memory in the Embedded Server,
- the file name of the last configuration (namespace) downloaded to the Embedded Server.

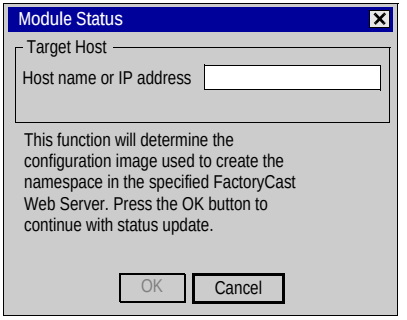
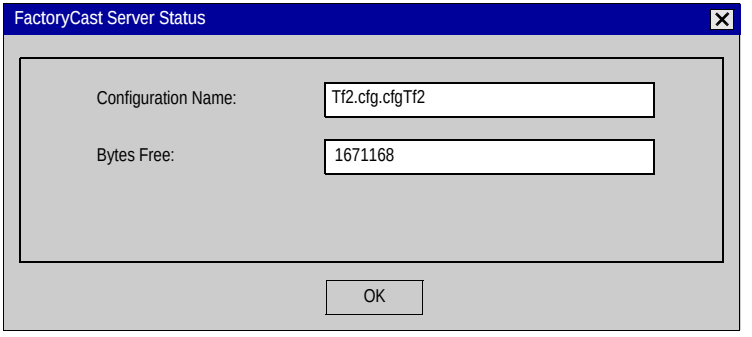
Memory Capacity

Consult the table below to find out how much memory your Embedded Server provides for customizing your Web site.

Embedded Server Model	Memory Available for Customizing the site
Quantum 140 NOE 2x1 10	Customizable based on plug-ins configured <1Mb
Premium TSX ETY 110 WS	Customizable based on plug-ins configured <2Mb
Quantum 140 NOE 771 1X	<8Mb
Premium TSX ETY 510	<8Mb
Micro TSX ETZ 510	<8Mb

Check Server Status

Follow the steps in the procedure below to check the server status.

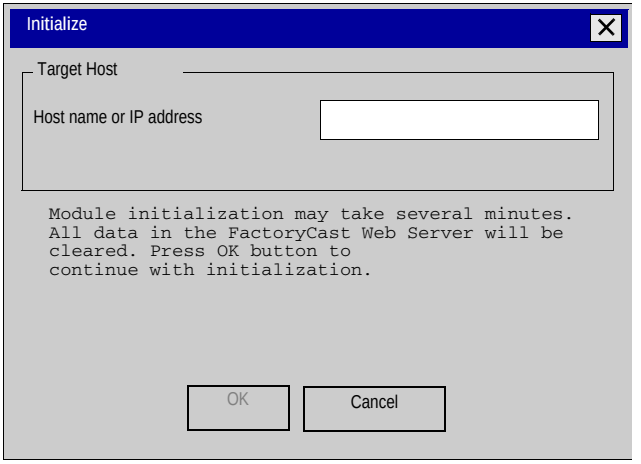
Step	Action
1	Select **Server | Status** from the menu bar. **Result:** The **Module Status** dialog box appears. 
2	In the **Host Name or IP Address** box, enter the Host Name or the IP Address of the Embedded Server.
3	Click **OK**. **Result:** The FactoryCast Server Status dialog box appears. It displays the current configuration file for the Embedded Server and the number of bytes free in memory. 
4	Click **OK** to exit the Module Status dialog box.

Restoring Module Defaults

Overview	If the server fails or its files become corrupted through user error, you can use the **Server | Restore Defaults** option to restore the FactoryCast Configurator files. This menu item restores the selected module to the state in which it was shipped from the factory. This may take some time, so be prepared.
Conditions for Restoring	In order to restore the Embedded Server to the factory supplied configuration, you must have unaltered FactoryCast Configurator files on your hard drive. If you have altered the FactoryCast Configurator directory files on your hard drive in any way—for instance, by replacing the default home page with your own home page—you must reinstall FactoryCast Configurator on your hard drive before you reflash the module. Otherwise, the module will be restored with the files on your hard drive which may have caused the initial problem.
Consequences of Restoring	Module initialization deletes all files on the FactoryCast Server except the data templates for the Data Editor and Graphic Editor displays.

Restore Defaults Command

When **Restore Defaults** is selected, the module will be cleared (except rdt and gdt directories) and the default plug-ins will be downloaded again. Follow the steps below.

Step	Action
1	Select **Server | Restore Defaults**. Result: The **Initialize** dialog box appears. 
2	Enter the **Host Name** or **IP Address** of the Embedded Server.
3	Click **OK**. **Note:** After a module has been initialized, it may be necessary to download the Plug-ins that are needed for your installation. (For a description of the available plug-ins, see *Loading the FactoryCast Applet Plug-ins, p. 187*.)

Setting the XWay Address

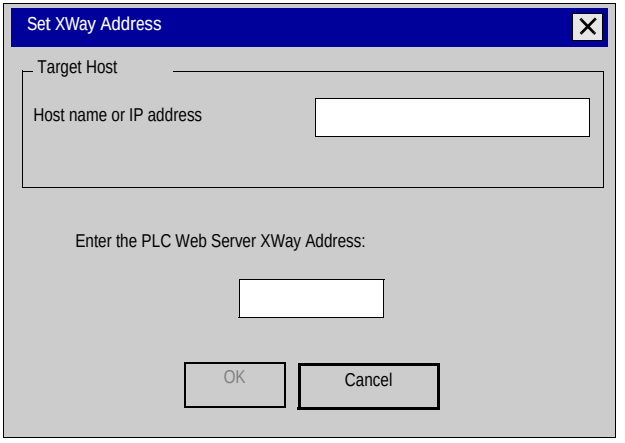
Overview

XWay is a Telemecanique communications protocol. If you are using the Premium ETY Embedded Server or the ETZ Embedded Server, you must set its XWay Address.

Accessing the XWay Address Dialog

Select **Server | Set XWay Address** from the menu bar.

Result: The **Set XWay Address** dialog box appears.



Setting the XWay Address

Enter the **Host Name** or **IP Address**, and the **XWay Address** of the Embedded Server.

The XWay address includes the address destination of the Premium ETY or Micro ETZ Embedded Server module and the address source of the PC. FactoryCast Client will scan a free source in this network from a network.63 nested 10 station address (net.63 to net.54). You must be careful to leave at least one address free in this range.

Note: The IP address of the Embedded Server and any host name should be assigned by your system administrator.

Applying the XWay Address

Click **OK** to download the XWay Address to the Premium ETY or the Micro ETZ Embedded Server.

Loading the FactoryCast Applet Plug-ins

Overview

Plug-ins are client components of FactoryCast that allow you to use system tools on the Diagnostics and Online Configurations Web page.

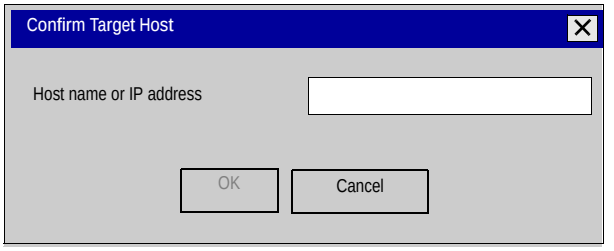
FactoryCast Plug-ins

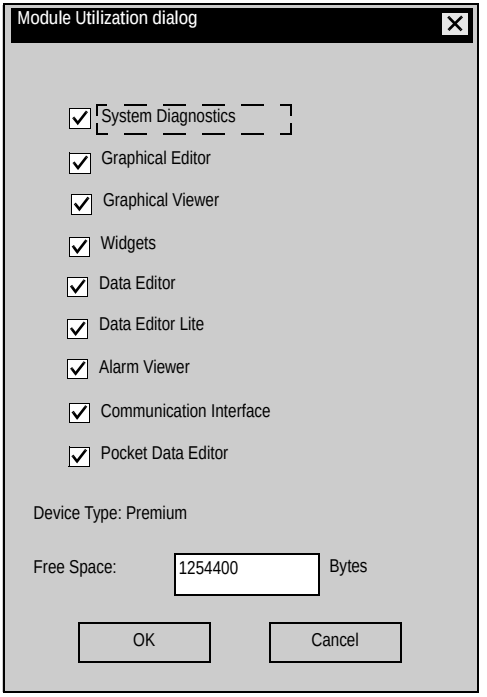
The following table describes each of the FactoryCast plug-ins.

Plug-in	Description
System Diagnostics	Gives the Embedded Server access to system status information such as for the I/O card, Controller, Ethernet, and Remote I/O health. This plug-in is installed when the Embedded Server is shipped.
Graphic Editor (See *Graphic Editor, p. 220*)	The Graphic Editor allows you to create and view data using Java Beans for visualization. Each graphic object can be linked to a symbol (variable) or direct address in the Embedded Server.
Graphic Viewer (See *Graphic Viewer, p. 268*)	The Graphic Viewer allows you to view data using JavaBeans for visualization. Each graphic object can be linked to a symbol (variable) or direct address in the Embedded Server.
Widgets	This plug-in lets you add graphical widget in your customized HTML pages.
Data Editor (See *Data Editor, p. 192*)	The Data Editor allows you to view and modify symbols (variables) and direct addresses which have been included in a namespace on the Embedded Server.
Data Editor Lite (See *Data Editor Lite, p. 209*)	The Data Editor Lite allows you to view and modify symbols (variables) and direct addresses which have been included in a namespace on the Embedded Server. It is a lighter version of Data Editor.
Alarm Viewer (See *Alarm Viewer, p. 337*)	When the PLC application has diagnostic properties activated, you can display the potential application faults with the Alarm Viewer.
Communication Interface	Allows the Embedded Server to access data from the controller. Must be installed in order for any other plug-in to be downloaded to the Embedded Server. This plug-in is installed when the Embedded Server is shipped.
Pocket Data Editor (See *Pocket Data Editor, p. 211*)	The Pocket Data Editor allows a Pocket PC user to view and modify symbols (variables) and direct addresses which have been included in a namespace on the Embedded Server.

Installing the FactoryCast Plug-ins

Follow these steps to install plug-ins:

Step	Action
1	Select **Server | Plug-ins** from the **Server** menu. Result: The **Confirm Target Host** dialog box appears. 
2	In the **Host Name** or **IP Address** box, enter the Host Name or the IP Address of the Embedded Server.

Step	Action
3	Click **OK**. **Result:** The FactoryCast Module Utilization dialog appears. This screen displays installed and uninstalled plug-ins on the **Embedded Server** along with the free space on the **Embedded Server**. 
4	Select the plug-ins that you want to install on the Embedded Server. Note: Graphical Viewer, Widgets, Data Editor and Pocket data Editor are available only on FactoryCast V3.1 Modules.
5	Click **OK**. An **Updating Plug-ins** box will appear to indicate that the plug-ins are being installed. When the box disappears, the update is complete.

Editors

8

Editors

Overview

This chapter describes the Data Editor and the Graphic Editor Java applets that enable you to create either dynamic data tables or dynamic graphic displays. Both editors are dynamically updated with run-time data from the PLC.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
8.1	Data Editor	192
8.2	Data Editor Lite	209
8.3	Pocket Data Editor	211
8.4	Graphic Editor	220
8.5	Graphic Viewer	268

8.1 Data Editor

Data Editor

Overview

This section describes how to use the Data Editor to view and modify the values of symbols (variables) and direct addresses.

What's in this Section?

This section contains the following topics:

Topic	Page
Data Editor	193
Data Editor Spreadsheet	196
Creating a Data Template	203
Saving a Data Template	204
Using an Existing Data Template	205
Inserting a Symbol (Variable) in a Data Template	206
Inserting a Direct Address in a Data Template	207
Modifying Data Values in a Data Template	208

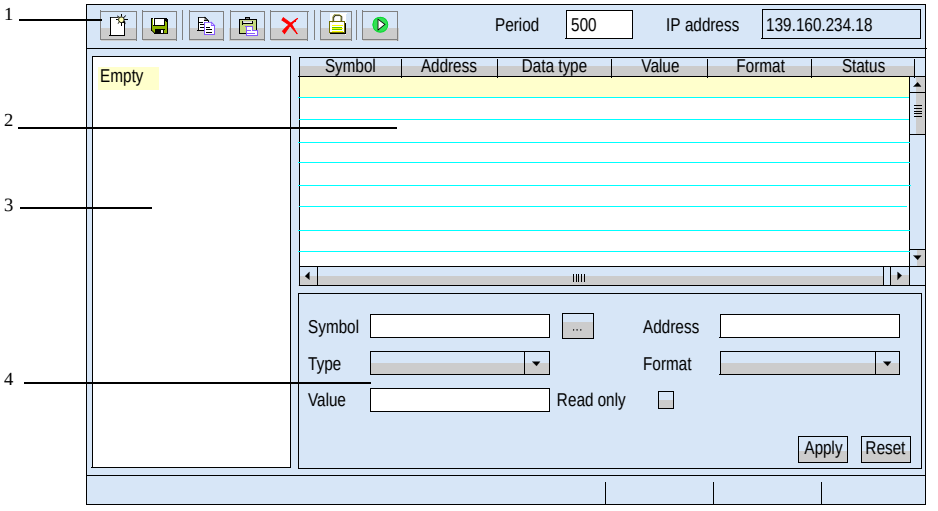
Data Editor

Overview

The Data Editor is a Java applet that enables you to create dynamic data tables that can be updated with run-time data from the PLC.

Elements of the Data Editor

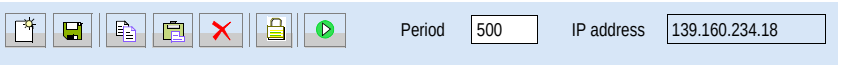
The Data Editor comprises 4 screen elements:



Number	Description
1	Tool bar (See *Tool bar, p. 195*).
2	Data Editor Template (See *Data Editor Spreadsheet, p. 196*). The data editor template is a spreadsheet that contains your data.
3	List of all data templates.
4	The Configuration Area makes it possible to: ● select (See *Inserting a Symbol (Variable) in a Data Template, p. 206*) and/or modify (See *Modifying Data Values in a Data Template, p. 208*) a symbol, ● modify (See *Modifying Data Values in a Data Template, p. 208*) a symbol value, ● select (See *Inserting a Direct Address in a Data Template, p. 207*) and/or modify an address, ● select the variable type, ● select the variable's display format, ● check the read-only option.

Tool bar

Here is a closer view of the buttons on the Data Editor's tool bar:



From left to right, the buttons indicate:

- create a new object,
- save the current application,
- copy the object,
- paste the object,
- delete the object,
- change the password,
- start or stop the animation,
- set the rate for polling the variable,
- display the server's IP address in format `<host>[:<:filename>]`.

Data Editor Spreadsheet

Overview

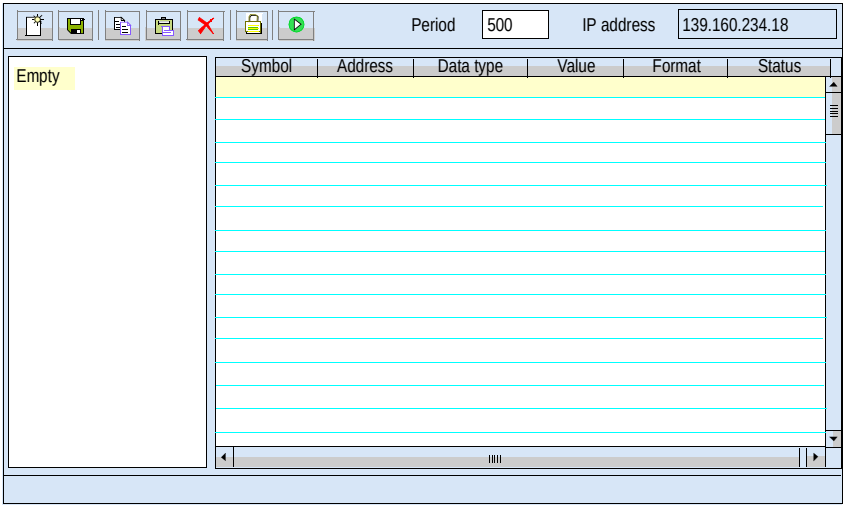
The Data Editor spreadsheet displays data with the following fields:

- **Symbol**,
- **Address**,
- **Data Type**,
- **Value**,
- **Format**,
- **Status**,

This topic provides a snapshot of the spreadsheet and an explanation of each field.

Spreadsheet

The figure shows the Data Editor spreadsheet:



Symbol Field

The **Symbol** column contains the names of Concept, PL7, or Unity Pro symbols (variables).

The only symbols (variables) that may be used in the Data Editor are those in the namespace on the Embedded Server.

Note: The Data Editor can only read values from a namespace that was created from the same program as the one running in the controller. The program used in the controller is displayed at the top of the Data Editor. If the namespace was created using a different program, its name is displayed at the bottom of the Data Editor.

Address Field

The Address column contains direct addresses and the addresses of Concept, PL7, or Unity Pro symbols (variables). Any direct address can be viewed by entering its reference in this field. It does not have to be included in the namespace.

Valid Direct Addresses for Quantum

Valid direct addresses for Quantum are:

- Coils (0x),
- Discreet inputs (1x),
- Input registers (3x),
- Output/holding registers (4x),
- Extended memory registers (6x).

For Unity Quantum PLC, direct addresses also include:

- %Mi (same as for 0X coils),
- %Ii (same as 1x for discreet inputs),
- %IWi (same as 3x for input registers),
- %MWi, %MDi, %MFi (same as 4x for holding registers).
  For a Unity Quantum PLC, a single bit of any "word address" (for example, %MWi, %IWi) can be specified by appending ".j" to the address, where "j" is a bit index in the range of 0 (LSB) to 15 (MSB). For example, bit 4 of the value at %MW101 would be specified as %MW101.4.
  Also for a Unity Quantum PLC, a direct address can include an index specification that allows it to be treated as an array variable. Indexed addressing can be used with a %Mi, %MWi, %MDi, or %MFi address by appending "[j]" to the address of the beginning of the array, where "j" is an unsigned integer value. For example, the third value of an array of float values starting at %MF201 would be specified as %MF201[2].

Valid Direct Addresses for Micro

The following table contains the valid direct addresses for Micro:

Address	Type	R(ead) or W(rite) Access
%KWi	WORD 16	R
%KDi	WORD 32	R
%MDi	WORD 32	R/W
%SDi	WORD 32	R/W
%Is.c	BOOLEAN	R
%Qs.c	BOOLEAN	R/W
%Mi	BOOLEAN	R/W
%Si	BOOLEAN	R/W
%MFi	REAL 32	R/W
%IWs.c.i	WORD 16	R
%MWi	WORD 16	R/W
%SWi	WORD 16	R/W
%QWs.c.i	WORD 16	R/W
%MWs.c.i	WORD 16	R/W
%MWs.MOD.i	WORD 16	R/W
%KWs.c.i	WORD 16	R/W
s=slot number, c=channel number, and i=range number		

Valid Direct Addresses for Premium

The following table contains the valid direct addresses for Premium, except Unity Premium:

Address	Type	R(ead) or W(rite) Access
%KWi	WORD 16	R
%KDi	WORD 32	R
%MDi	WORD 32	R/W
%SDi	WORD 32	R/W
%Irs.c	BOOLEAN	R
%Qrs.c	BOOLEAN	R/W
%Mi	BOOLEAN	R/W
%Si	BOOLEAN	R/W
%MFi	REAL 32	R/W
%IWrs.c.i	WORD 16	R
%MWi	WORD 16	R/W
%SWi	WORD 16	R/W
%QWrs.c.i	WORD 16	R/W
%MWrs.c.i	WORD 16	R/W
%MWrs.MOD.i	WORD 16	R/W
%KWrs.c.i	WORD 16	R/W
r = rack number, s = slot number, c = channel number, and i = range number		

Valid Direct Addresses for Unity Premium

The following table contains the valid direct addresses for Unity Premium:

Address	Type	R(ead) or W(rite) Access
%KWi	WORD 16	R
%KDi	WORD 32	R
%MDi	WORD 32	R/W
%SDi	WORD 32	R/W
%Ir.s.c	BOOLEAN	R
%Qr.s.c	BOOLEAN	R/W
%Mi	BOOLEAN	R/W
%Si	BOOLEAN	R/W
%MFi	REAL 32	R/W
%IWr.s.c.	WORD 16	R
%MWi	WORD 16	R/W
%SWi	WORD 16	R/W
%QWr.s.c.	WORD 16	R/W
%MWr.s.c.i	WORD 16	R/W
%KWr.s.c.i	WORD 16	R/W
r = rack number, s = slot number, c = channel number, and i = range number		

The following table contains the valid FIP I/O addresses for Premium, except Unity Premium:

Address	Type	R(ead) or W(rite) Access
%I\rs.2.d\m.c	BOOLEAN	R
%Q\rs.2.d\m.c	BOOLEAN	R/W
%IW\rs.2.d\m.c.i	WORD 16	R
%QW\rs.2.d\m.c.i	WORD 16	R/W
%MW\rs.2.d\m.c.i	WORD 16	R/W
%KW\rs.2.d\m.c.i	WORD 16	R
r = rack number, s = slot number, d = device number, m = module number, c = channel number and i = range number		

The following table contains the valid FIP I/O addresses for Unity Premium:

Address	Type	R(ead) or W(rite) Access
%I\bs.cp\0.m.c	BOOLEAN	R
%Q\bs.cp\0.m.c	BOOLEAN	R/W
%IW\bs.cp\0.m.c.i	WORD 16	R

Address	Type	R(ead) or W(rite) Access
%QW\bs.cp\0.m.c.i	WORD 16	R/W
%MW\bs.cp\0.m.c.i	WORD 16	R/W
%KW\bs.cp\0.m.c.i	WORD 16	R
bs = bus number, cp = connection point, m = module number, c = channel number and i = range number		

For Unity Premium, a single bit of any "word address" (for example, %MWi, %SWi, %KWi) can be specified by appending ".j" to the address, where "j" is a bit index in the range of 0 (least significant bit) to 15 (most significant bit). For example, bit 4 of the value at %MW101 would be specified as %MW101.4.

Also for Unity Premium PLC, a direct address can include an index specification that allows it to be treated as an array variable. Indexed addressing can be used with a %Mi, %MWi, %MDi, %MFi, %KWi or %KD address by appending "[j]" to the address of the beginning of the array, where "[j]"is an unsigned integer value. For example, the third value of an array of float values starting at %MF201 would be specified as %MF201[2].

Data Type Field

The Data Type field contains the data type of the symbol (variable) or direct address. Symbol (variable) data types appear automatically when the symbol (variable) is located. Direct address data types must be set by the user from a drop-down list.

The following data types are valid:

Abbreviation	Data Type
INT	16-bit signed integer
UINT	16-bit unsigned integer
DINT	32-bit signed integer
UDINT	32-bit unsigned integer
REAL	32-bit IEEE floating point
TIME	32-bit unsigned integer (in ms)
DATE	Date (32-bit BCD)
TOD	Time of Day (32-bit BCD)
BOOL	1-bit discreet (boolean)

Value Field

The Value column is filled with the value of the symbol (variable) or direct address. This field is updated continuously.

Format Field

The Format field contains the format type for displaying the value of the symbol (variable) or direct address. The following formats are available:

Abbreviation	Format Type
bool	Boolean
dec	Decimal
hex	Hexadecimal
binary	Binary
ASCII	bytes displayed as ASCII characters
time	day_hr_min_sec_ms
date	YYYY-MM-DD or HH:MM:SS

Status Field

The Status column contains messages about the status of communications with the symbol (variable) or direct address. If communications are normal, the status message will be "OK."

If there is a problem communicating with the symbol (variable) or direct address, the **Status** column will contain an error message describing the problem.

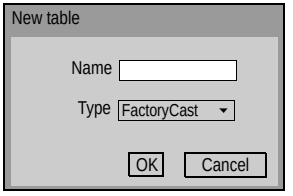
Creating a Data Template

Overview

To display some symbols (variables) or direct addresses, you must create a new data template.

Creating a Data Template

Follow the steps in the table below to create a data template:

Step	Description
1	If you would like a new one, select the  button on the tool bar. Result: The **New table** box appears. 
2	Enter the name of the new Data template.
3	Click **OK**.

Note: Save (See *Saving a Data Template, p. 204*) the current spreadsheet before selecting a new spreadsheet. (Selecting a new spreadsheet deletes the current spreadsheet.)

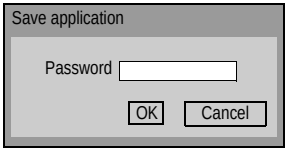
Saving a Data Template

Overview

If you save a data template, you can use it again to view or modify the same symbols (variables) or direct addresses.

Saving a Data Template

Follow the steps in the table below to save a data template.

Step	Action
1	Click the  button on the tool bar. Result: The **Save application** box appears. 
2	Enter the password (USER by default).
3	Click **OK**.

Using an Existing Data Template

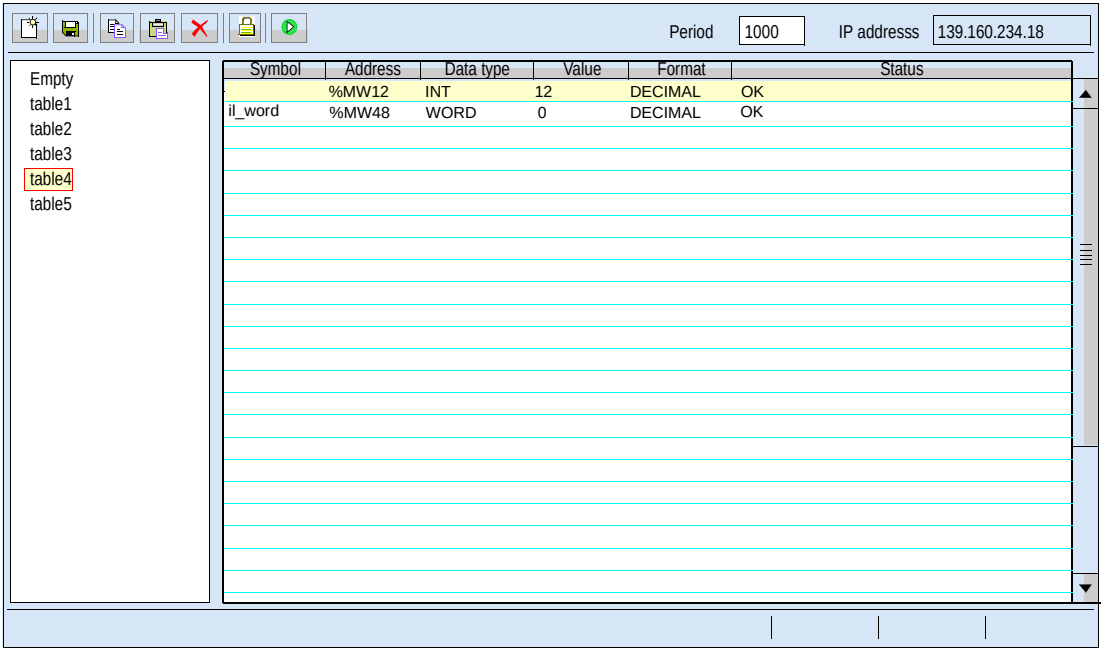
Overview

After you save a data template, you can use it to view or modify the values of the same symbols (variables) and direct addresses.

Finding a Data Template

A treeview menu on the Data Editor lists all the saved data templates.

DATA EDITOR



Retrieving a Data Template

Select the data template you want from the treeview menu. It will appear on a spreadsheet.

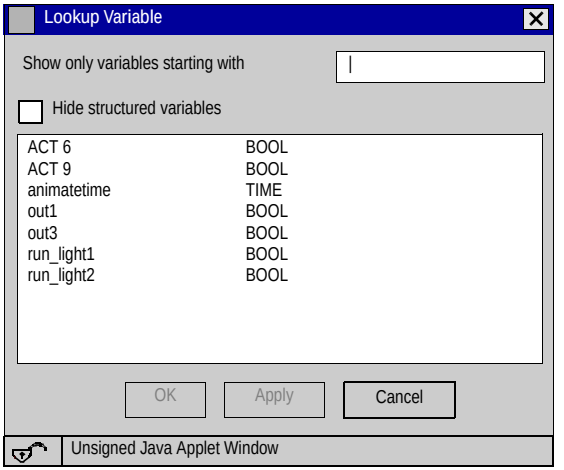
Inserting a Symbol (Variable) in a Data Template

Overview

If you want to view or modify the value of a symbol (variable) in the namespace, you must insert that symbol (variable) in a data template.

Inserting a Symbol (Variable)

Follow the steps in the table below to insert a Symbol (variable):

Step	Action
1	Double-click on an empty row in the spreadsheet. **Result**: The data editor's configuration area appears.
2	In the configuration area, click on the  button. **Result:** The **Lookup Variable** dialog box appears. 
3	Select the symbol (variable) you want to insert in the data template.
4	Click **OK**. **Result:** The symbol (variable) you selected is displayed in the **Symbol** field.
5	In the configuration area, click on **Apply**. **Result:** A new row corresponding to the symbol (variable) you selected is displayed in the spreadsheet.

Inserting a Direct Address in a Data Template

Overview

If you want to view or modify the value of a direct address, you must insert that direct address in a data template.

How to Insert a Direct Address

Follow the steps in the table below to insert a Direct address:

Step	Action
1	Double-click on an empty row in the spreadsheet. **Result**: The data editor's configuration area appears.
2	In the **Address** field of the configuration area, enter the variable's Direct address.
3	In the configuration area, click on **Apply**. **Result:** A new row that corresponds to the variable address is displayed in the spreadsheet.

Modifying Data Values in a Data Template

Overview

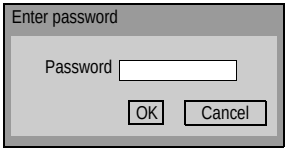
You can use the Data Editor to modify the value of a symbol (variable) and of a direct address, and send the new value(s) to the controller.

Restrictions on Modifying Data

You can only modify the value of a symbol (variable) or of a direct address that is write-enabled in the namespace.

Modifying Data

Follow the steps in the table below to modify data:

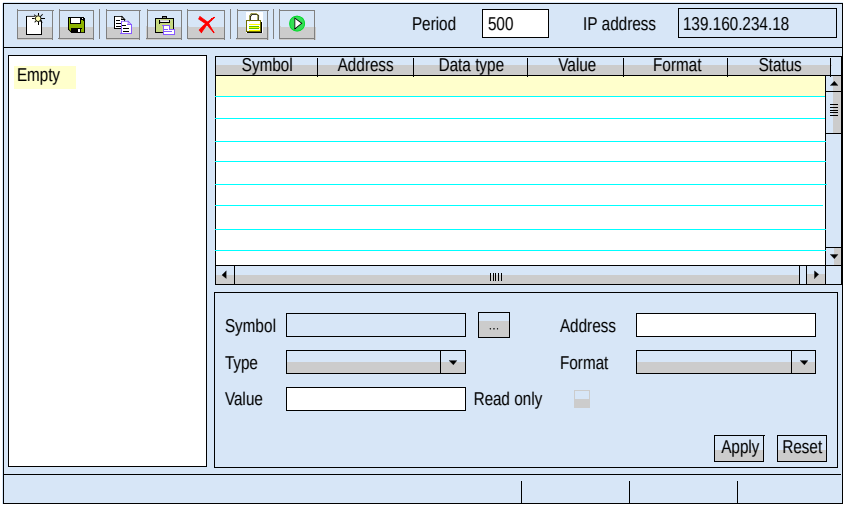
Step	Action
1	Double-click on a spreadsheet row that corresponds to the symbol (variable). **Result**: The data editor's configuration area appears.
2	In the **Value** field of the configuration area, change the value of the symbol (variable) or direct address.
3	Click **Apply**.
4	**Result:** A **Password** dialog appears. 
5	Enter the password (USER by default).
6	Click **OK** **Result:** The new value is sent to the controller.

8.2 Data Editor Lite

Data Editor Lite

Overview

Data Editor Lite is a version of the Data Editor that is smaller in size and thus faster to download, especially for use via a dialup connection. It provides the same interface.



Variables

Data Editor Lite accepts the following IEC and Concept variables:

Address	Type	Display
%MW IEC internal word	INT	DECIMAL
%MD IEC double word	DINT	DECIMAL
%M IEC internal bits	BOOL	BOOLEAN
400000 Concept variable	INT or DINT	DECIMAL
000000 Concept variable	BOOL	BOOLEAN

Tables

Data Editor Lite can reuse the same tables created with Data Editor and Pocket Data Editor. However, Data Editor tables use a wider range of variable types than Data Editor Lite. When Data Editor Lite encounters a variable it cannot manage, it displays `Not Supported`. In this case, the variable cannot be edited.

Operating Mode

Refer to the Operating Mode section for Data Editor (See *Data Editor, p. 192*).

Note: You cannot insert symbols (variables) in a Data Template with Data Editor Lite, you can only insert direct addresses.

8.3 Pocket Data Editor

Pocket Data Editor

Overview

This section describes the Pocket Data Editor, which runs on Pocket PC.

What's in this Section?

This section contains the following topics:

Topic	Page
Pocket Data Editor	212
Using Pocket Data Editor	214

Pocket Data Editor

Overview

Pocket Data Editor is a version of the Remote Data Editor that operates in a Pocket PC environment. A Pocket PC with a WiFi connection can be used as a mobile client station to access data on remote equipment. Pocket Data Editor supports remote data monitoring, remote control, and remote diagnostics.

Pocket PC Requirements

- only Pocket PCs with ARM processors,
- Pocket PC Windows 2003 mobile edition.

Pocket PC Windows 2002 mobile edition is not supported.

Capabilities

Pocket Data Editor can open, edit, and save the same data tables as the PC version in a format appropriate for Pocket PCs.

Note: If you have developed custom pages for your Web server, Internet Explorer for Pocket PC can access the same custom HTML pages on an equipment Web site. A Pocket PC screen is smaller than one on a PC, so the Web site must be adapted to return the appropriate page based on the client's browser type.

Availability

Pocket Data Editor is available on the FactoryCast v3.1 Web server. It can be used with the following modules:

- a TSX ETY 5103 module for Premium,
- a TSX ETZ 510 module for Micro,
- an NOE 771 11 module for Quantum.

Note: Pocket Data Editor is disabled when the Pocket PC is connected to and synchronized with a PC via USB using the ActivSync tool.

Note: Some communication problems can arise between a Pocket PC and the FactoryCast module when the WiFi (802.11) power save mode is activated on a Pocket PC. See your Pocket PC user manual for the procedure to disable this feature.

Variables

Pocket Data Editor accepts the following variables:

Address	Type	Display
%MW IEC internal word	INT	DECIMAL
%MD IEC double word	DINT	DECIMAL
%M IEC internal bits	BOOL	BOOLEAN
400000 Concept variable	INT or DINT	DECIMAL
000000 Concept variable	BOOL	BOOLEAN

Warning

WARNING

Pocket Data Editor is not a safety product

You must provide an acceptable level of safety in your application in accordance with the process.

Failure to follow this instruction can result in death or serious injury.

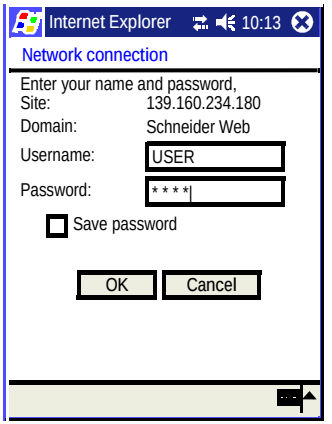
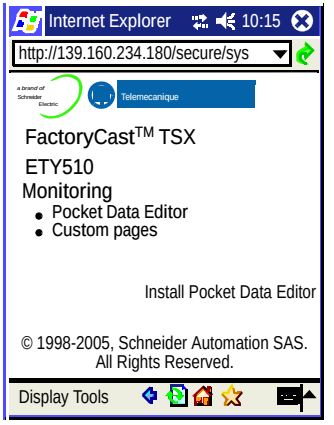
Using Pocket Data Editor

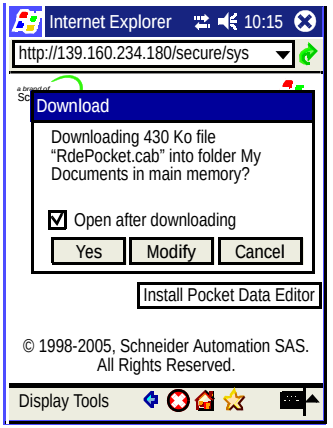
Overview

When you first connect a Pocket PC to the FactoryCast Web server, you must download and install the Pocket Data Editor software on your Pocket PC. After this, you can launch the Pocket Data Editor by clicking on the **Pocket Data Editor** link on the equipment Web site home page.

Installing Pocket Data Editor

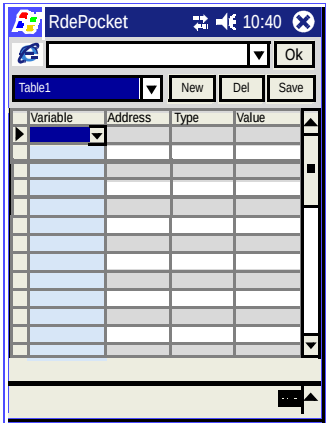
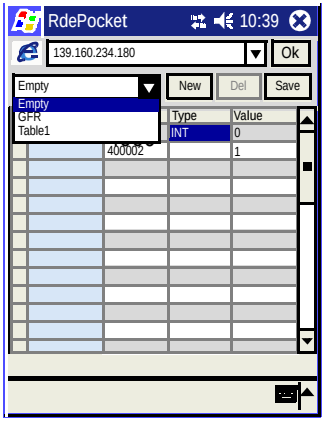
The following table describes the installation of the Pocket Data Editor:

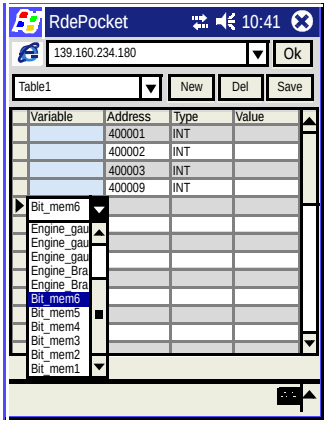
Step	Action
1	Turn on the Pocket PC.
2	Launch Internet Explorer .
3	Enter the IP address of the remote equipment. **Result**: The following window appears: 
4	Enter the username and password. (The defaults are USER and USER.) **Result**: the FactoryCast Server home page appears: 

Step	Action
5	Click on **Install Pocket Data Editor**. **Result**: The following window appears: 
6	Click on **YES**.

Using Pocket Data Editor

The following table describes the use of Pocket Data Editor

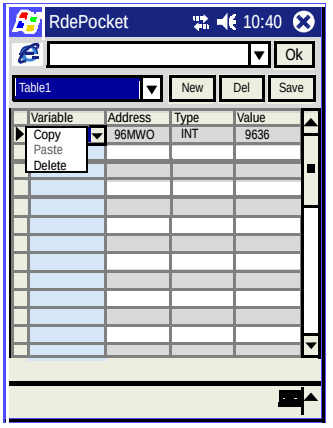
Step	Action
1	Click on the **Pocket Data Editor** link on the equipment Web site home page. **Result**: The following window appears:  Push the **New** button to create a new table after entering its name. Push the **Del** button to delete the current table. Push the **Save** button to save the new version of the table after entering the HTTP password.
2	Select the desired table via the drop-down menu:  The Pocket PC version can use the same tables defined in the Data Editor for PC and in Data Editor Lite. Not all variables are accessible. (Restrictions are detailed at the Variables discussion (See *Variables, p. 213*).) A selected variable may be reported as **not supported** in the status bar. For an unsupported variable type, the string ??? appears in the value cell.

Step	Action
3	If you want to view or modify the value of a PLC variable, insert the variable symbol or address in the Pocket Data table.  You can add new addresses, following the restrictions detailed in the Variables section (See *Variables, p. 213*). To modify a PLC variable, select the corresponding **value** cell and enter the new value.

Copy, Paste, or Delete Table Rows

To copy, paste, or delete, select one or more rows by clicking in the first cell (to the left of the **variable** cell). The selected rows are highlighted. Hold down the mouse button in this cell until a popup menu appears, then select the desired action.

- To copy, select one or more rows that have values (that are not empty).
- To paste the copied rows, select a row, empty or not. If several rows are copied, the other rows are pasted one after the other.
- To delete one or more rows, select rows that have values (that are not empty) and choose **delete**.



Using customized pages

If you want to display a custom page, click on the **Custom pages** link on the equipment Web site home page.

Launch Pocket Data Editor from its icon

Pocket Data Editor can also be launched directly from its program icon.

8.4 Graphic Editor

Graphic Editor

Overview

This section describes the functions and features of the Graphic Editor, a Java applet that enables you to create dynamic graphic displays via a Web browser, using a set of pre-defined graphic objects. The Graphic Editor is only used as an editor for creating and modifying displays. The Graphic Viewer is the run-time environment for viewing displays while they are dynamically animated with run-time data from the PLC. This viewer is lighter than the editor, resulting in faster loading times and operation.

What's in this Section?

This section contains the following topics:

Topic	Page
Graphic Editor Overview	221
Top Window User Functions	223
Display Window User Functions	228
Property Sheet	231
Security	233
Graphic Editor Applet Parameters	234
Graphic Objects	236
Extended Graphic Objects	257

Graphic Editor Overview

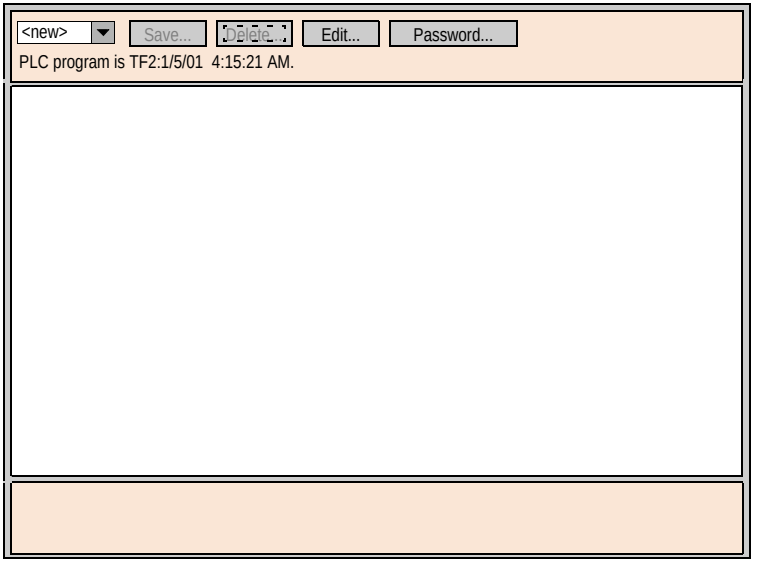
The Interface

The Graphic Editor applet is separated into three windows:

- Top Window: This window provides the area for presenting all the user controls and dialogs for creating, saving, reading, and editing a graphic display.
- Display Window: This window provides an area for presenting the current graphic display. When you create a new graphic display, this window becomes an "empty canvas" on which you can add the graphic objects that will compse the desired graphic display.
- Message Window: This window presents any messages generated by the Graphic Editor.

Illustration

The figure below shows the Graphic Editor applet with its initial top window, and empty display and message windows.



Note: Owing to memory restrictions, you can not download both the Data Editor and Graphic Editor on the 140 NOE 211 10 and NOE 251 10 FactoryCast Servers. To download both, you must remove an equivalent amount of plug-ins.

Graphic Objects

All graphic objects provided with the Graphic Editor are capable of communicating with the PLC from which the Graphic Editor applet was downloaded. There is no additional "wiring" of the graphic objects with "communication objects." All graphic objects are standalone, meaning there are no connections required between them and each is capable of operating independently.

Viewing a Graphic Display

After the Graphic Editor applet has been uploaded to a Web browser, you will usually want to either view a graphic display (for monitoring/controlling the PLC application) or create or modify a graphic display. A user who only wants to view and interact with existing graphic displays (e.g., an operator) can select the Graphic Viewer link instead of Graphic Editor. They will see a window with the widgets that does not include the Edit menu. This viewer loads faster than the standard Graphic Editor because it is lighter. You need only to enter a password to write data to the PLC.

Create and Modify Graphic Displays

To create and modify graphic displays, click the **Edit...** button to see the standard graphic editor functions. These include selecting objects from a palette, dropping them onto a canvas, moving and resizing them with a mouse, and setting object properties. You can immediately test the modified graphic display with run-time data from the PLC by clicking the **Done** button to exit edit mode. When satisfied with your creation, the graphic display can be saved to the PLC for re-use by clicking the **Save...** button, assuming you entered the correct password.

User Functions

Most of the Graphic Editor's user functions are available as Top Window User Functions (See *Top Window User Functions, p. 223*). From the display window, you can directly manipulate a graphic object's size and location. All properties of a graphic object (e.g., scaling values, labels, colors, PLC addresses of the run-time data) are set in the Property Sheet (See *Property Sheet, p. 231*).

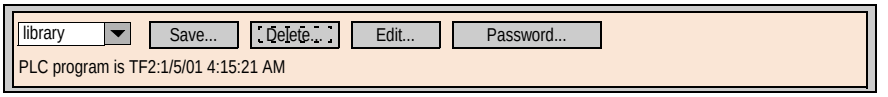
Top Window User Functions

Overview

The Graphic Editor applet's top window consists of several "dialog panels," only one of which is visible at any time. Switching from one dialog to another is done by clicking buttons on the current dialog. This topic describes the dialog panels that compose the top window.

Top Dialog

The **Top Dialog** is the dialog panel that is initially shown in the top window when the Graphic Editor applet is started. Access to all other dialog panels of the top window is from this dialog.

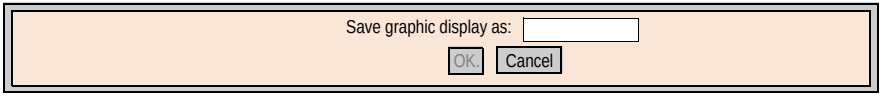


The controls of the **Top Dialog** provide the following functions:

- **Drop-down List**. The drop-down list box shows all graphic display files that have been saved to the Web server module and are available for retrieval. When you select a graphic display from this list, the graphic display currently in the window is replaced with the selected one. If the current graphic display has been modified since it was last saved, you are asked for confirmation that the changes are to be discarded. If the special entry <new> is chosen from the list, the display window is cleared and a new graphic display can be created.
- **Save**. The **Save** button makes the **Save Dialog** visible. This button is disabled until you have entered a correct write-enabled password.
- **Delete**. The **Delete...** button makes the **Delete Dialog** visible. This button is disabled until you have entered a correct password, or if the current graphic display has not yet been saved.
- **Edit**. The **Edit...** button makes the **Edit Dialog** visible.
- **Password**. The **Password...** button makes the **Password Dialog** visible.
- **Information display area**. The information display area shows the name and version of the Concept, PL7, or Unity Pro program that is running in the connected PLC.

Save Dialog

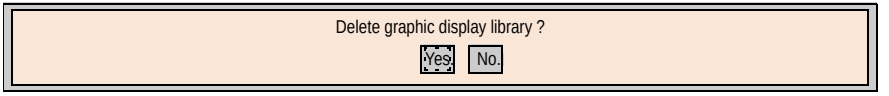
The **Save Dialog** allows you to save the current graphic display.



When the **Save Dialog** is presented, the name of the current graphic display is shown in the dialog's text field. If the current graphic display has never been saved (i.e., a "new" graphic display), then the text field is blank. Once you have either accepted the current name (with a "save" operation) or provided a new name (with a "save as" operation), then you can click the **OK** button to save the contents of the current graphic display to the Web server module. The **Cancel** button will cause the **Top Dialog** to be shown again, with no action being taken.

Delete Dialog

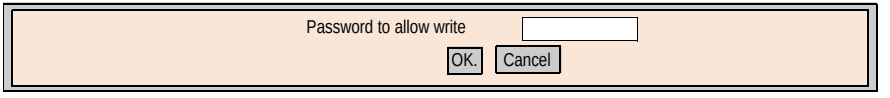
The **Delete Dialog** allows you to delete the current graphic display.



If you click **Yes**, the existing graphic display window is cleared and the graphics file on the Web server module is deleted. Clicking **No** will cause the **Top Dialog** to be shown again, with no action being taken.

Password Dialog

The **Password Dialog** allows you to enter the password that enables those user functions that modify graphic display files or PLC run-time data values.

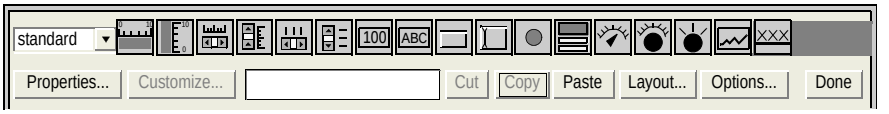


If you enter the correct password and click **OK**, then you will be allowed to save and delete the current graphic display. Correct entry of the password also permits you to write new values to the PLC (via those graphic objects that support writing values to a PLC, if any). Clicking **OK** when the text field is empty clears the current password permissions (if there are any). The **Cancel** button redisplays the **Top Dialog** without changing the current password permissions.

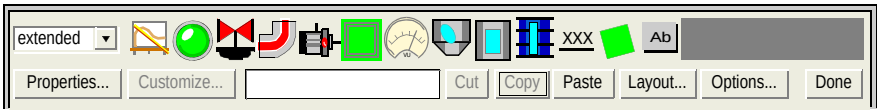
Edit Dialog

The **Edit Dialog** allows you to select a graphic object for placement in the display window, and provides access to all graphic editing functions. The available graphic objects are presented in a set of palettes, with one palette visible at a time. There are two palettes.

The standard palatte:



The extended palatte:

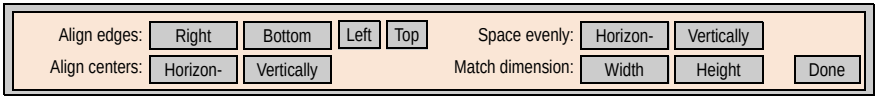


The controls of the **Edit Dialog** provide the following functions:

- The **Drop-down List Box** shows the set of available palettes. When you select the name of a palette from the list, the graphic objects in that palette appear in the palette display area of the dialog.
- The **Palette** shows the graphic objects in the current palette. An icon depicts each graphic object's type (meter, button, etc.). When you click any icon in the palette, a graphic object of the corresponding type is selected for insertion. If you click in an open area of the display window while the Graphic Editor is in "insert mode," an instance of the selected graphic object is inserted into the graphic display.
- The **Information Area** shows the name and size of the graphic object that is currently selected.
- The **Cut** button causes the currently selected graphic object(s) to be removed from the graphic display and saved to a buffer (i.e., an internal clipboard), replacing the existing contents of the buffer.
- The **Copy** button causes the currently selected graphic object(s) to be copied to the buffer, replacing the existing contents of the buffer.
- The **Paste** button causes the content of the clipboard to be inserted into the upper left corner of the graphic display. The pasted graphic objects can then be moved to the desired location in the display.
- The **Properties** button displays the currently selected graphic object's Property Sheet (See *Property Sheet, p. 231*).
- The **Customize** button displays the currently selected object's Customizer (See *Customizing Complex Graphic Objects, p. 230*) (if the graphic object has one).
- The **Layout** button makes the **Layout Dialog** visible.
- The **Options** button makes the **Options Dialog** visible.
- The **Done** button makes the **Top Dialog** visible again.

Layout Dialog

The **Layout Dialog** allows you to change the position and size of a group of graphic objects.



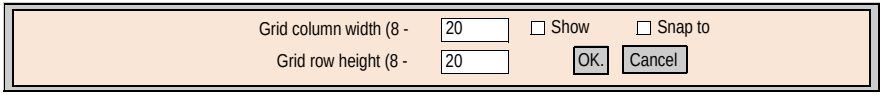
The controls of the **Layout Dialog** provide the following functions:

- For aligning the edges of graphic objects, the **Right**, **Bottom**, **Left**, and **Top** buttons move the selected graphic objects so that their specified sides are at the same position. At least two graphic objects must be selected for these buttons to be enabled.
- For aligning the center lines of graphic objects, the **Horizontally** and **Vertically** buttons move the selected graphic objects so that their horizontal or vertical center lines, respectively, are at the same position. At least two graphic objects must be selected for these buttons to be enabled.
- For positioning graphic objects so that they are evenly spaced, the **Horizontally** and **Vertically** buttons move the selected graphic objects so that either the horizontal or vertical spacing between the objects is the same. At least three graphic objects must be selected for these buttons to be enabled.
- To automatically size graphic objects, use the **Width** and **Height** buttons to re-size the currently selected graphic objects so that either the widths or heights, respectively, of the objects match. At least two graphic objects must be selected for these buttons to be enabled.
- The **Done** button makes the **Edit Dialog** visible again.

Note: For all layout operations (except **Space evenly**) one of the selected objects is considered the "reference object" to which all other selected objects adjust for their new position or dimension. For example, when the "Width" button is pressed, all selected objects have their widths changed to match that of the reference object. The reference object is differentiated from the other selected objects by making its selection box a different color than the others.

Options Dialog

The **Options Dialog** allows you to change the settings related to a grid that can be drawn in the display window. The grid is solely for assistance in editing a graphic display and is shown only when the Graphic Editor is in "edit mode." Edit mode starts when you switch to the **Edit Dialog** and ends when you return to the **Top Dialog**.



The **Options Dialog** controls provide the following functions:

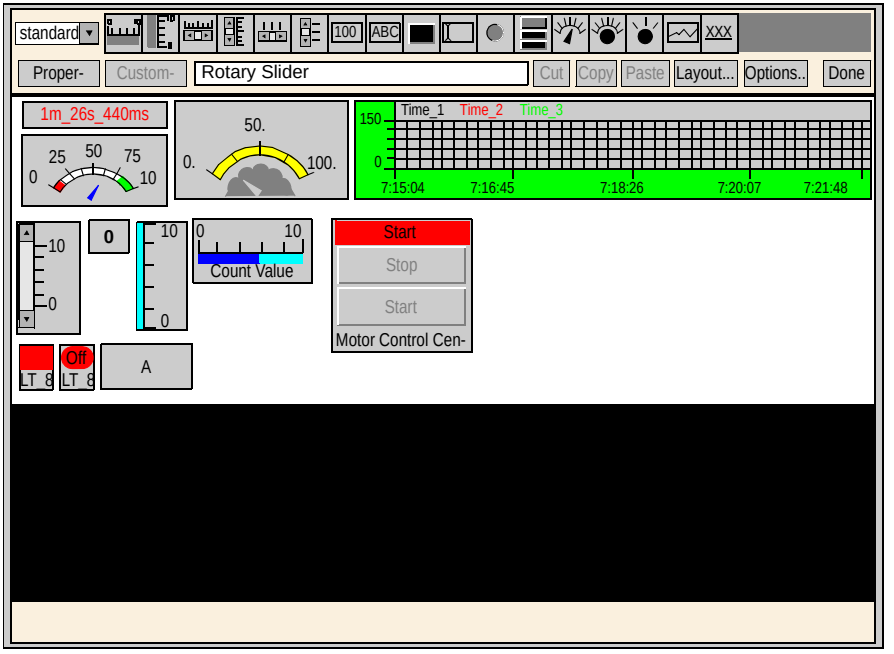
- The cell size of the grid can be changed by the entering the grid's column width and row height in the dialog's text fields.
- If the **Show grid** check-box is checked, the grid is drawn; otherwise, no grid is shown.
- If the **Snap to grid** check-box is checked, then, when you change the size or position of a graphic object, the changed coordinate(s) or dimension(s) is automatically adjusted to coincide with a grid point.
- The **OK** button causes the current option settings to become active, and the **Edit Dialog** to be shown again.
- The **Cancel** button causes the **Edit Dialog** to be shown again, with no option settings being changed.

Display Window User Functions

Overview

The user functions available from the **Graphic Editor** display window allow you to select, move, and size objects. Moving and sizing operations require that your first select those graphic object(s) that you want to modify. A selected object is indicated by its surrounding selection box; an unselected or deselected object has no surrounding selection box.

The figure below shows the **Graphic Editor** display.



Selecting Graphic Objects

A graphic object's selection state (selected/deselected) can be set via the following user actions:

- A single graphic object can be selected by simply clicking on it with a mouse. If any other objects are currently selected, they will be deselected.
- Multiple graphic objects can be selected with a selection box in the display window. If you press a mouse button in an open area of the display window (i.e., not on a graphic object) and drag the mouse without releasing it, you will see a dotted outline box. One corner of the box is fixed where the mouse button was initially pressed while the opposite corner tracks the current mouse position. When the mouse button is released, all objects that intersect the selection box are selected. Objects outside the box will be deselected.
- A graphic object's selection state can be toggled between selected and deselected without affecting the selection state of other objects by pressing the CTRL key when clicking on the object. With this action, graphic objects can be individually added or removed from the group of selected objects.
- A graphic object can be selected without affecting the selection state of any other objects by pressing the SHIFT key when clicking on the object. When an object is selected this way, it becomes the *reference object* (see Layout Dialog *Top Window User Functions, p. 223*) for the group of selected objects. The primary purpose of this action is to change the reference object in a group of selected objects prior to invoking one of the **Layout** operations.
- All graphic objects can be deselected by clicking the mouse in an open area of the display window, that is, not on a graphic object.

Sizing Graphic Objects

A graphic object's size can be changed by first selecting it, then using the mouse to change the size of the object's selection box. As you move the mouse over an object's selection box, the mouse pointer changes to reflect the type of sizing operation to be performed. If you press a mouse button while the mouse is over an object's selection box and drag the mouse without releasing it, a dotted outline box appears. When the mouse button is released, the object's size is changed to match the size of the outline. There are eight possible sizing actions depending on which part of an object's selection box is dragged. Each corner of the box will allow only its adjacent sides to move; each side of the box will allow only that side to move.

Moving Graphic Objects

A graphic object can be moved in the display window with the mouse. If you press a mouse button while the cursor is over an object and drag without releasing the button, then a selection box will be shown. When the mouse button is released, the object moves to the location of the selection box.

Multiple graphic objects can be moved by first selecting the objects to be moved, and then dragging the entire group of objects in the same way a single object is moved. While a group of objects is moved, a selection box is shown for each object in the group.

Setting Graphic Object Properties

You can set a graphic object's properties via the Property Sheet (See *Property Sheet, p. 231*). If the Property Sheet is visible, the properties of the selected graphic object are presented for editing. The Property Sheet can displayed by pressing the **Properties...** button or by double-clicking the mouse anywhere on the selected object in the display window.

Customizing Complex Graphic Objects

Some complex graphic objects have a very large number of properties. Configuring such an object with the Property Sheet can be cumbersome. An available Customizer can ease the configuration of complex graphic objects. A Customizer is a dialog window designed specifically to configure its associated graphic object. When the Graphic Editor detects that a selected graphic object has a Customizer, it will enable the **Customizer...** button, which brings up the graphic object's Customizer. When you double-click on a graphic object that is associated with a Customizer, the Customizer comes up (instead of the Property Sheet). If a graphic object is associated with a Customizer, the only item in the Property Sheet is its name.

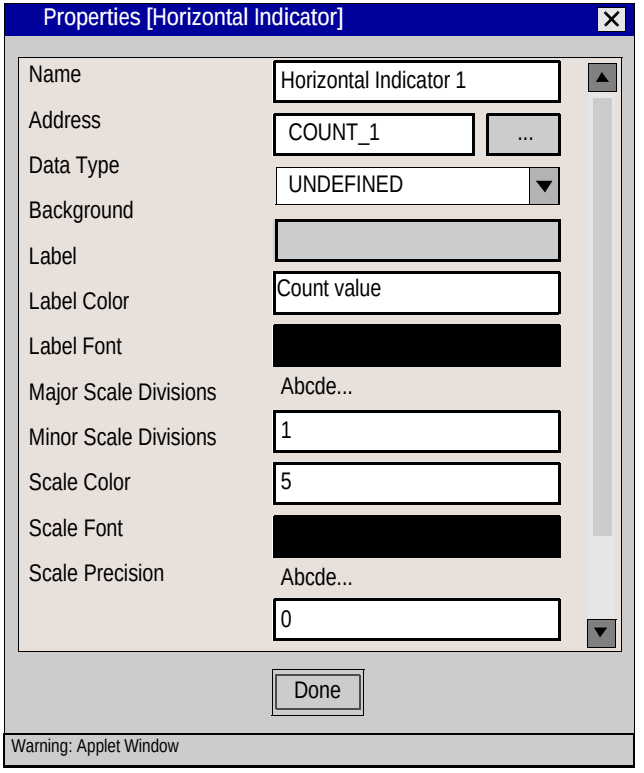
Display Background Image

A Graphic Editor display has a **Background Image** property that can be used to designate an image as the display's background. The image can be a GIF or JPEG file. All file locations are relative to the Embedded Server's /wwwroot directory. For example, if the image "cool.gif" was put in the Embedded Server's /wwwroot/images directory, then the Background Image property should be set to /images/cool.gif.

Property Sheet

Overview

The Property Sheet is a "floating" (non-modal) dialog that shows the configurable properties of the currently selected graphic object:

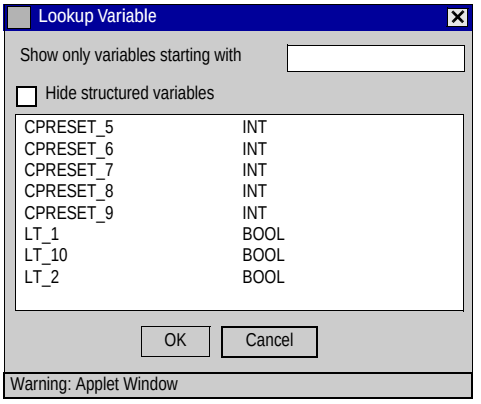


The properties of a graphic object are specific to an object's type. The properties are presented in a scrollable list, with the name and the value of each property listed. The Graphic Editor comes with a description of graphic objects (See *Graphic Objects, p. 236*).

Lookup Dialog

For each of the graphic objects provided with the Graphic Editor, a property editor is provided for its **Address** property. This editor not only allows you to directly enter the address of a Quantum/Premium/Micro register (or Concept/PL7/Unity Pro variable name), but also provides access to the **Lookup Dialog**. The Lookup Dialog allows you to pick a Concept/PL7/Unity Pro symbol (variable) name from a list of symbol (variables) that have been "Web enabled" by the FactoryCast Configurator.

The following figure shows the **Lookup Variable** dialog box:



Security

Security

You are protected from unauthorized access to your PLC data in three ways.

- The HTML page containing the Graphic Editor applet has been placed in a "secure" directory on the Web module, then the Web browser user is asked for a password before being allowed to download the HTML page.
- You must enter the correct password via the **Password Dialog** to have permission to save/delete files or send data values to the connected PLC. With respect to sending data values to the PLC, the Graphic Editor will enforce its "read-only" mode by disabling the user-input controls of all graphic objects.
- The FactoryCast Configurator allows you to specify that an item is read-only. The **Graphic Editor** will enforce the read-only attribute of a symbol (variable) or address by rejecting any request to set a new value for the data item, and informing the user in the **Graphic Editor** message window.

Graphic Editor Applet Parameters

Overview

Three applet parameters can customize the behavior of the **Graphic Editor**. Applet parameters are specified with <PARAM> tags within the <APPLET> tag in the Graphic Editor's HTML page. The parameters recognized by the **Graphic Editor** applet are:

- **LOAD**: This parameter tells the **Graphic Editor** to auto-load a specific graphics file when it starts. If the file does not exist, an error message appears. If this parameter is not provided in the <APPLET> tag, then a file is not auto-loaded at startup and you have to select an initial graphics file from the list provided by the **Graphic Editor**.
- **MODE**: This parameter tells the **Graphic Editor** whether to startup in its normal "Edit Mode" or in a special "View Mode." When started in view mode, the **Graphic Editor** shows only its display window. When this parameter is used with the LOAD parameter, a Web site can be designed using HTML pages that are dedicated to specific graphic displays. No explicit selection of graphic files is required by a user, providing more typical HMI screen behavior. The possible values for this parameter are:
  - **EDIT** (default value): The **Graphic Editor** starts up in its normal Edit Mode.
  - **VIEW_RO**: The **Graphic Editor** starts up in read-only view mode. The Web browser user will not be allowed to send data values to the PLC.
  - **VIEW_RW**: The **Graphic Editor** starts up in read/write view mode. The Web browser user will be allowed to send data values to the PLC after entering the write-access password.
- **AUTO_LOGIN**: This parameter tells the **Graphic Editor** to automatically enter the password that is required to permit writing to the PLC. If the **MODE** parameter is set to **VIEW_RW** or **EDIT**, then setting this parameter to **TRUE** will cause the **Graphic Editor** to allow writing to the PLC without requiring the user to enter the password. The possible values for this parameter are **FALSE** (default) and **TRUE**.

Example

The following is an example of an applet tag for the **Graphic Editor** that causes it to start up in view mode while automatically loading a graphics file named **UNIT_1**. In this case, the Web browser allows you to send values to the PLC via any graphic objects that support sending values (assuming you have entered the write-access password).

```
<APPLET codebase="/classes"
archive="SAComm.jar,GDE.jar,Widgets.jar"
code="com.schneiderautomation.gde.GdeApplet"
width="700" height="514">
<PARAM name="LOAD" value="UNIT_1">
<PARAM name="MODE" value="VIEW_RW">
<PARAM name="AUTO_LOGIN" value="FALSE">
</APPLET>
```

Graphic Objects

Overview

The set of graphic objects provided with the **Graphic Editor** supports the construction of graphic displays that mimic conventional instrument panels. All of the data monitoring and control objects have built-in communication capabilities and are designed as standalone graphic objects.

Additionally, each object in the **Graphic Editor** set is available in an applet version to support customers that want to put several simple applets on a single HTML page. When used in conjunction with `LiveBeanApplet`, the **Graphic Editor** graphic objects can be used in the same way as the `LiveLabelApplet`.

This topic describes standard graphic objects and their properties.

Horizontal Indicator

A Horizontal Indicator provides an analog representation of the value of a symbol (variable) or direct address in a PLC. It is a horizontal bar that represents the value as a percentage of its range in engineering units. Optionally, a digital indication of the value can be shown in the center of the bar area.

The following table describes the properties for the Horizontal Indicator:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 256*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 256*
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used on the label	
Major Scale Divisions	The number of major (labeled) scale divisions	0 to 100
Minor Scale Divisions	The number of minor (unlabeled) scale divisions	0 to 100
Scale Color	The color of the scale and its labels	
Scale Font	The font used on scale labels	
Scale Precision	The number of fractional digits to be shown for scale labels (set to -1 to use a general exponential format)	-1 to 6
Maximum EU Value	The maximum value, in engineering units, of the direct address or symbol (variable)	
Minimum EU Value	The minimum value, in engineering units, of the direct address or symbol (variable)	
Maximum PLC Value	The maximum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Minimum PLC Value	The minimum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Value Visible	Indicates whether a digital display of the scaled value is to be shown	
Value Font	The font for the digital display of the value (if shown)	
Bar Background	The background color of the bar indicator area	

Property	Description	Limits
Bar Color	The color of the indicator bar (when the scaled value within High/Low limits)	
High High Limit Value	The value in engineering units for the ‘High High’ limit	
High High Limit Color	The color of the indicator bar when the scaled value is greater than the ‘High High’ limit	
High Limit Value	The value in engineering units for the ‘High’ limit	
High Limit Color	The color of the indicator bar when the scaled value is greater than the ‘High’ limit	
Low Limit Value	The value in engineering units for the ‘Low’ limit	
Low Limit Color	The color of the indicator bar when the scaled value is less than the ‘Low’ limit	
Low Low Limit Value	The value in engineering units for the ‘Low Low’ limit	
Low Low Limit Color	The color of the indicator bar when the scaled value is less than the ‘Low Low’ limit	
Limit Deadband	The deadband (as percentage of EU range) to apply to High/Low limit checking	0 to 10
Border Width	The width (in pixels) of the graphic object’s border	0 to 32
Border Color	The color of the graphic object’s border	
PLC Value	A simulated, raw (unscaled) input value for testing the graphic object	See Note 3, *Notes, p. 256*

Vertical Indicator

A Vertical Indicator provides an analog representation of the value of a symbol (variable) or direct address in a PLC. It is a vertical bar that represents the value as a percentage of its range in engineering units.

The following table describes the properties of the Vertical Indicator:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 256*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 256*
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Major Scale Divisions	The number of major (labeled) scale divisions	0 to 100
Minor Scale Divisions	The number of minor (unlabeled) scale divisions	0 to 100
Scale Color	The color of the scale and its labels	
Scale Font	The font used for scale labels	
Scale Precision	The number of fractional digits to be shown for scale labels (set to -1 to use a general exponential format)	-1 to 6
Maximum EU Value	The maximum value, in engineering units, of the direct address or symbol (variable)	
Minimum EU Value	The minimum value, in engineering units, of the direct address or symbol (variable)	
Maximum PLC Value	The maximum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Minimum PLC Value	The minimum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Bar Background	The background color of the bar indicator area	
Bar Color	The color of the indicator bar (when the scaled value within High/Low limits)	
High High Limit Value	The value in engineering units for the ‘High High’ limit	
High High Limit Color	The color of the indicator bar when the scaled value is greater than the ‘High High’ limit	

Property	Description	Limits
High Limit Value	The value of the ‘High’ limit in engineering units	
High Limit Color	The color of the indicator bar when scaled value is greater than the ‘High’ limit	
Low Limit Value	The value of the ‘Low’ limit in engineering units	
Low Limit Color	The color of the indicator bar when the scaled value is less than the ‘Low’ limit	
Low Low Limit Value	The value of the ‘Low Low’ limit in engineering units	
Low Low Limit Color	The color of the indicator bar when the scaled value is less than the ‘Low Low’ limit	
Limit Deadband	The deadband (as percentage of EU range) to apply to High/Low limit checking	0 to 10
Border Width	The width (in pixels) of the graphic object’s border	0 to 32
Border Color	The color of the graphic object’s border	
PLC Value	A simulated, raw (unscaled) input value for testing the graphic object	See Note 3, *Notes, p. 256*

Horizontal or Vertical Slider

A Horizontal or Vertical Slider provides an analog representation of the value of a symbol (variable) or direct address in a PLC. It is a scroll bar with a "thumb" position that represents the value as a percentage of its range in engineering units. With the mouse, you can change the value of the scroll bar, sending a new value to the PLC.

The following table describes the properties for the Horizontal or Vertical Slider:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 256*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 256*
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Major Scale Divisions	The number of major (labeled) scale divisions	0 to 100
Minor Scale Divisions	The number of minor (unlabeled) scale divisions	0 to 100
Scale Color	The color of the scale and its labels	
Scale Font	The font used for scale labels	
Scale Precision	The number of fractional digits to be shown for scale labels (set to -1 to use a general exponential format)	-1 to 6
Maximum EU Value	The maximum value, in engineering units, of the direct address or symbol (variable)	
Minimum EU Value	The minimum value, in engineering units, of the direct address or symbol (variable)	
Maximum PLC Value	The maximum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Minimum PLC Value	The minimum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Block Increment	The amount that the scaled value should change when the scroll bar's scroll area is clicked	
Unit Increment	The amount that the scaled value should change when the scroll bar's arrow buttons are clicked	
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	

Horizontal or Vertical Selector

A Horizontal or Vertical Selector allows you select from a set of choices. When a selection is made, the value corresponding to the choice is sent to the PLC. The choices are shown as labels of a "scale," with the current selection indicated by the position of the "thumb" of a scroll bar.

The following table describes the properties of the Horizontal or Vertical Selector:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address (or the name of a symbol (variable)) to monitor	See Note 1, *Notes, p. 256*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 256*
Background	The background color for the graphic object	
Choices	The choices for the selector. Each choice is given as a ‘label=value’ entry (when you select ‘label,’ ‘value’ is sent to PLC).	Minimum of two choices required
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Scale Visible	Indicates whether a "scale," labeled with the choices, is to be shown	
Scale Color	The color of the scale and its labels	
Scale Font	The font used for scale labels	
Border Width	The width (in pixels) of the graphic object’s border	0 to 32
Border Color	The color of the graphic object’s border	

Digital Indicator

A Digital Indicator provides a numeric representation of the value of a symbol (variable) or direct address in a PLC. The value can be shown in various formats, and can be made to change color when a preset high or low limit is exceeded.

The following table describes the properties of the Digital Indicator:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 256*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 256*
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Value Format	The format (decimal, hex, etc.) to use in displaying the scaled value	
Value Precision	The number of fractional digits to be shown for the scaled value (set to -1 to use a general exponential format)	-1 to 6
Value Background	The background color of the value display area	
Value Color	The text color for the digital display of the value	
Value Font	The font used for the digital display of the value	
Units	The label for the value's engineering units (appended to the numerical display of the value)	
Maximum EU Value	The maximum value, in engineering units, of the direct address or symbol (variable)	
Minimum EU Value	The minimum value, in engineering units, of the direct address or symbol (variable)	
Maximum PLC Value	The maximum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Minimum PLC Value	The minimum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
High High Limit Value	The value of the 'High High' limit in engineering units	
High High Limit Color	The color of the indicator bar when the scaled value is greater than the 'High High' limit	
High Limit Value	The value of the 'High' limit in engineering units	

Property	Description	Limits
High Limit Color	The color of the indicator bar when the scaled value is greater than the ‘High’ limit	
Low Limit Value	The value of the ‘Low’ limit in engineering units	
Low Limit Color	The color for the indicator bar when scaled value is less than the ‘Low’ limit	
Low Low Limit Value	The value of the ‘Low Low’ limit in engineering units	
Low Low Limit Color	The color of the indicator bar when the scaled value is less than the ‘Low Low’ limit	
Limit Deadband	The deadband (as percentage of EU range) to apply to High/Low limit checking	0 to 10
Border Width	The width (in pixels) of the graphic object’s border	0 to 32
Border Color	The color of the graphic object’s border	
PLC Value	A simulated, raw (unscaled) input value for testing the graphic object	See Note 3

Message Display

A Message Display shows a text message based on the value of a symbol (variable) or direct address in a PLC. For each specified message, a specified value triggers its display.

The following table describes the properties of the Message Display:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 256*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 256*
Background	The background color of the graphic object	
Messages	The set of messages to display. Each message is given as a 'value=text' entry (when the PLC value equals 'value,' 'text' is displayed as the message)	Minimum of one message required
Message Background	The background color of the message display area	
Message Color	The color of the message text	
Message Font	The font used for the message text	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	
PLC Value	A simulated input value for testing the graphic object	See Note 3, *Notes, p. 256*

Push Button

A Push Button allows you to send preset value(s) to a PLC when clicked with the mouse.

The following table describes the properties of the Push Button:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 256*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 256*
Background	The background color of the graphic object	
Values	The value(s) to send to the PLC	See Note 4, *Notes, p. 256*
Reset Values	The value(s) to send to the PLC after the reset delay time has expired. If no reset values are provided, no reset action will occur.	
Reset Delay	The delay time (in milliseconds) that the Push Button should wait after sending the value(s) to the PLC before sending the reset value(s)	0-2000
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Button Label	The text label for the button	
Button Background	The color of the button	0 to 100
Button Label Color	The color of the button label	
Button Label Font	The font used for the button label	
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	

Direct Output Station

The Direct Output Station allows you to enter a numeric value in a text field directly with the keyboard. When the entered value is within preset high and low limits, a **Set** button is enabled. In this case, the entered value will be sent to the PLC when you press either the **Set** button or the ENTER key (if the input field has keyboard input focus).

The following table describes the properties of the Direct Output Station:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 256*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 256*
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Maximum EU Value	The maximum value, in engineering units, of the direct address or symbol (variable)	
Minimum EU Value	The minimum value, in engineering units, of the direct address or symbol (variable)	
Maximum PLC Value	The maximum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Minimum PLC Value	The minimum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Maximum Input	The maximum value, in engineering units, that is valid for the entered input value	
Minimum Input	The minimum value, in engineering units, that is valid for the entered input value	
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	

Indicator Light

The Indicator Light provides a dual-state indication of the value of a symbol (variable) or direct address in a PLC. Unless the **Input Inverted** property is set to **TRUE**, an input value of zero is deemed **OFF** and a non-zero value is deemed **ON**. If the **Flash Interval** property is set to greater than zero, the light will flash while the input value is on.

The following table describes the properties of the Indicator Light:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address (or the name of a symbol (variable)) to monitor	See Note 1, *Notes, p. 256*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 256*
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Off Word	The text to show when the input value is off	
Off Word Background	The background color of the light when the **Off Word** is shown	
Off Word Color	The color of the **Off Word** text	
Off Word Font	The font used for the **Off Word** text	
On Word	The text to show when the input value is on	
On Word Background	The background color of the light when the **On Word** is shown	
On Word Color	The color of the **On Word** font	
On Word Font	The font used for the **On Word** text	
Flash Interval	The flashing time period (in milliseconds) of the light when the input value is on. Set to zero for no flashing.	200 to 2000
Shape	The shape (circle, rectangle, etc.) of the light	
Input Inverted	If **TRUE**, inverts the input value (light will show the **Off Word** when input value is on).	
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	
PLC Value	A simulated input value for testing the graphic object	See Note 3, *Notes, p. 256*

Motor Control Station

The Motor Control Station is designed to mimic the typical start/stop push button station that is often used to control motors. This graphic object is essentially a composite of two push buttons and an indicator light. A Customizer is provided to make it easier to set the object's many properties. All of the properties (except Name) are set with its Customizer, not with the **Graphic Editor's** Property Sheet.

The following table describes the properties of the Motor Control Station:

Property	Description	Limits
Name	The name of the graphic object	
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	
Indicator Light	Same properties as the Indicator Light graphic object, excluding the shared properties listed above	
Top Push Button	Same properties as the Push Button graphic object, excluding the shared properties listed above	
Bottom Push Button	Same properties as the Push Button graphic object, excluding the shared properties listed above	

Analog Meter

An Analog Meter provides an analog representation of the value of a symbol (variable) or direct address in a PLC. It is represented as a pointer on a circular dial; its position indicates the value as a percentage of its range in engineering units. You can set the size of the meter's circular dial (degrees sweep of a circle), the colors of the dial, and the style of the pointer.

The following table describes the properties of the Analog Meter:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 256*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 256*
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Major Scale Divisions	The number of major (labeled) scale divisions	0 to 100
Minor Scale Divisions	The number of minor (unlabeled) scale divisions	0 to 100
Scale Color	The color of the scale and its labels	
Scale Font	The font used for scale labels	
Scale Precision	The number of fractional digits to be shown for scale labels (set to -1 to use a general exponential format)	-1 to 6
Maximum EU Value	The maximum value, in engineering units, of the direct address or symbol (variable)	
Minimum EU Value	The minimum value, in engineering units, of the direct address or symbol (variable)	
Maximum PLC Value	The maximum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Minimum PLC Value	The minimum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Dial Degrees Sweep	The amount of a circular arc to use for drawing the dial	60 to 300
Pointer Type	The type (needle, arrow head, etc.) of pointer to use	
Pointer Color	The color for the pointer	

Property	Description	Limits
Dial Color	The color of the dial (that part that is within the High/Low limits)	
High High Limit Value	The value of the 'High High' limit in engineering units	
High High Limit Color	The color of the indicator bar when the scaled value is greater than the 'High High' limit	
High Limit Value	The value of the 'High' limit in engineering units	
High Limit Color	The color of the indicator bar when the scaled value is greater than the 'High' limit	
Low Limit Value	The value of the 'Low' limit in engineering units	
Low Limit Color	The color of the indicator bar when the scaled value is less than the 'Low' limit	
Low Low Limit Value	The value of the 'Low Low' limit in engineering units	
Low Low Limit Color	The color of the indicator bar when the scaled value is less than the 'Low Low' limit	
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	
PLC Value	A simulated, raw (unscaled) input value for testing the graphic object	See Note 3, *Notes, p. 256*

Rotary Slider

A Rotary Slider provides an analog representation of the value of a symbol (variable) or direct address in a PLC. It is represented as a knob on a circular dial; its position indicates the value as a percentage of its range in engineering units. The dial size and knob color can be set. With the mouse, you can change the position of the knob, sending a new value to the PLC.

The following table describes the properties of the Rotary Slider:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 256*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 256*
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Major Scale Divisions	The number of major (labeled) scale divisions	0 to 100
Minor Scale Divisions	The number of minor (unlabeled) scale divisions	0 to 100
Scale Color	The color of the scale and its labels	
Scale Font	The font used for scale labels	
Scale Precision	The number of fractional digits to be shown for scale labels (set to -1 to use a general exponential format)	-1 to 6
Dial Degrees Sweep	The amount of a circular arc to use for drawing the dial	60 to 300
Dial Color	The color of the dial	
Knob Color	The color used for the knob	
Maximum EU Value	The maximum value, in engineering units, of the direct address or symbol (variable)	
Minimum EU Value	The minimum value, in engineering units, of the direct address or symbol (variable)	
Maximum PLC Value	The maximum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Minimum PLC Value	The minimum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Border Width	The width (in pixels) of the graphic object's border	0 to 32

Property	Description	Limits
Border Color	The color of the graphic object's border	

Rotary Selector

A Rotary Selector allows you to select from a set of choices. When a selection is made, the value corresponding to the choice is sent to the PLC. The choices are shown as labels of a "scale," with the current selection indicated by the position of the knob. The size of the circular dial (degrees sweep of a circle) and knob color can be set.

The following table describes the properties of the Rotary Selector:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 256*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 256*
Background	The background color of the graphic object	
Choices	The choices for the selector. Each choice is given as a 'label=value' entry. (When you select 'label,' 'value' is sent to PLC.)	Minimum of two choices required
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Scale Visible	Indicates whether a "scale," labeled with the choices, is to be shown	
Scale Color	The color of the scale and its labels	
Scale Font	The font used for scale labels	
Dial Degrees Sweep	The amount of a circular arc to use for drawing the dial	60 to 300
Knob Color	The color of the knob	
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	

Trend Recorder

A Trend Recorder provides a continuous, time-based charting of the value of up to six symbol(s) (variables) or direct addresses in a PLC. A Trend Recorder emulates a strip-chart recorder, with the pens on the right, and the "paper" moving from right to left. A vertical scale can be shown on the left side of the chart for showing the range of the values being recorded, and a horizontal scale can be shown below the chart for showing the time span of the chart. You can set the rate at which the chart is updated, and the appearance of the chart.

A Customizer is provided to make it easier to set this object's many properties. All properties (except Name) are set with its Customizer, not with the **Graphic Editor's** Property Sheet.

The following table describes properties of the Trend Recorder. Properties available for each pen are described in the next table:

Property	Description	Limits
Name	The name of the graphic object	
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Major Scale Divisions	The number of major (labeled) scale divisions	0 to 100
Minor Scale Divisions	The number of minor (unlabeled) scale divisions	0 to 100
Scale Color	The color of the scale and its labels	
Scale Font	The font used for scale labels	
Scale Precision	The number of fractional digits to be shown for scale labels (set to -1 to use a general exponential format)	-1 to 6
Maximum EU Value	The maximum value, in engineering units, of the direct address or symbol (variable)	
Minimum EU Value	The minimum value, in engineering units, of the direct address or symbol (variable)	
Update Period	The update interval (in seconds) for the chart	0.5 to 120
Time Scale Divisions	The number of horizontal scale divisions	0 to 6
Chart Background	The color of the chart area	
Grid Color	The color of the grid drawn in the chart area	
Vertical Grid Divisions	The number of vertical divisions for the grid	0 to 100

Property	Description	Limits
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	

These Trend Recorder properties are available for each pen:

Property	Description	Limits
Address	The direct address (or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 256*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 256*
Maximum PLC Value	The maximum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Minimum PLC Value	The minimum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 256*
Pen Color	The color of the "pen" used to record the scaled value	
Pen Label	The label used to identify the pen	

Display Link

A Display Link is a special graphic object that allows you to switch to another graphic display by clicking on it with a mouse. To indicate that the object is a link to another display, the text label for the link is underlined and the mouse cursor changes to a hand icon when the mouse is moved over it. This object is especially useful when the **Graphic Editor** is used in **view mode**, where no drop-down list of graphic displays is available for selecting a display.

A Display Link can also be used as a hyperlink to an HTML file. If a URL is entered as the **Link Display Name**, the URL can be opened in a new browser window if you press the SHIFT key while clicking the link; otherwise, the existing browser window is replaced with the URL when the link is clicked.

If the **Link Display Name** is blank, then the **Label** will be shown as not underlined, and the displayed object becomes a simple text label.

The following table describes the properties of the Display Link:

Property	Description	Limits
Label	The label of the link	
Link Display Name	The name of the graphic display to be loaded when the link is clicked, or a URL of a Web page	
Label Color	The color of the label	
Label Font	The font used for the label	

Notes

These are the notes for this topic:

1.	If the Address property of a graphic object is a direct address, the Data Type property is set to UNDEFINED, a default Data Type (BOOL, INT, DINT, or REAL, based on the implied size of the data value) is used. If the Address property is a symbol (variable) name, the Data Type property does not have to be specified and can be set to UNDEFINED. If, however, the Data Type property is specified for a symbol (variable), it must exactly match the symbol's (variable's) actual data type. If the Address property is a direct address for a discreet PLC reference (Quantum 0x/1x reference), the Data Type property must be set to BOOL. The Data Type property may be set to BOOL only for a discreet PLC reference.
2.	The meanings of the possible values of the Data Type property are:

Data Type	**Meaning**
UNDEFINED	no data type specified
BOOL	1-bit discreet (Boolean)
SHORT	8-bit signed integer
USHORT	8-bit unsigned integer
INT	16-bit signed integer
UINT	16-bit unsigned integer
DINT	32-bit signed integer
UDINT	32-bit unsigned integer
REAL	32-bit IEEE floating point
TIME	32-bit unsigned integer (in milliseconds)
DATE	Date (32-bit BCD)
TOD	Time of Day (32-bit BCD)
DT	Date and Time (64-bit BCD)

3.	The limits for the Maximum PLC Value and Minimum PLC Value properties are the natural limits of the Data Type property that is set. A Data Type setting of UNDEFINED is treated as REAL with respect to its limit values.
4.	For a Push Button, at least one value must be provided. If the Address property is a symbol (variable) name, then only one value will ever be sent to the PLC, and any additional values are ignored. If the Address property is a direct address, then all of the values provided will be sent to the PLC as an array of values starting at the specified direct address.

Extended Graphic Objects

Overview

The set of extended graphic objects provided in the Graphic Editor is intended to support building graphic displays that mimic advanced graphic panels. All of the data monitoring and control objects have built-in communication capabilities and are designed as standalone graphic objects.

Additionally, to support customers that want to put several simple applets on a single HTML page, each object in the Graphic Editor set is provided in an applet version. When used in conjunction with the `LiveBeanApplet`, Graphic Editor graphic objects can be used in the same way as the `LiveLabelApplet`.

ASCII Text Writer

The ASCII text writer is based on the message display widget. It allows you to input new text.

The properties of the ASCII text writer are:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 267*
Max. Text Length	The maximum length of the text	
Text Color	The color of the text	
Text Font	The font of the text	
Swap Bytes	False if target order of bytes is same as PC one	
Value	The text itself	

Bar Graph

A bar graph provides an analog representation of the value of a symbol (variable) or direct address in a PLC. It draws a vertical bar whose length is proportional to the value as a percentage of its range in engineering units.

The properties of the bar graph are:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 267*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 267*
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used on the label	
Maximum EU Value	The maximum value, in engineering units, of the direct address or symbol (variable)	
Minimum EU Value	The minimum value, in engineering units, of the direct address or symbol (variable)	
Maximum PLC Value	The maximum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 267*
Minimum PLC Value	The minimum raw (unscaled) value of the direct address or symbol (variable) in the PLC	See Note 3, *Notes, p. 267*
Bar Background	The background color of the bar indicator area	
Bar Color	The color of the indicator bar (when scaled value within High/Low limits)	
High High Limit Value	The value of the High High limit in engineering units	
High High Limit Color	The color of the indicator bar when the scaled value is greater than the High High limit	
High Limit Value	The value of the High limit in engineering units	
High Limit Color	The color of the indicator bar when scaled value is greater than the High limit	
Low Limit Value	The value of the Low limit in engineering units	
Low Limit Color	The color of the indicator bar when the scaled value is less than the Low limit	
Low Low Limit Value	The value of the Low Low limit in engineering units	

Property	Description	Limits
Low Low Limit Color	The color of the indicator bar when the scaled value is less than the Low Low limit	
Limit Deadband	The deadband (as percentage of EU range) to apply to High/Low limit checking	0 to 10
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	
PLC Value	A simulated, raw (unscaled) input value for testing the graphic object	See Note 3, *Notes, p. 267*

Bitmap

The bitmap widget displays a static bitmap on the screen.

The properties of the bitmap widget are:

Property	Description	Limits
Name	The name of the graphic object	
Background	The background color of the graphic object	See Note 1, *Notes, p. 267*
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Border Width	The width (in pixels) of the graphic object's border	
Border Color	The color of the graphic object's border	
Bitmap Choices	Represents the filenames of the custom bitmaps to display. The default root path of the file location directory is **/FLASH1/wwwroot;**; **":images/** thus refers to **/FLASH1/wwwroot/images/** .	

Generic Bitmap

The generic bitmap widget lets you display one static bitmap for each distinct value of a PLC variable. It can be used to display dynamic animations, for instance the changing level in a tank.

The properties of the Generic Bitmap widget are:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 267*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 267*
Background	The background color of the graphic object	See Note 1, *Notes, p. 267*
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Bitmap Choices	Represents the filenames of the custom bitmaps to display. Clicking on this property opens a text editor that makes it possible to type the PLC value conditions and related bitmaps to display, such as "0:key.gif:images/" where 0 is the PLC value, "key.gif" the bitmap file related to the value, "images" the directory in which the file is located. The default root path of the file location directory is **/FLASH1/wwwroot**; **images/** thus refers to **/FLASH1/wwwroot/images/**.	
Border Width	The width (in pixels) of the graphic object's border	
Border Color	The color of the graphic object's border	
PLC Value	A simulated input value for testing the graphic object behavior.	

Graphic Link

A graphic link is a special graphic object that lets you switch to another graphic display by clicking on it. Graphic links can also be recognized by their underlined labels, and the mouse cursor changes to a hand icon when the mouse moves over them. This object is especially useful when the Graphic Editor is used in view mode, where no drop-down list of graphic displays is available.

A graphic link can also be used as a hyperlink to an HTML file. If a URL is entered as the **Link Display Name**, the URL can be opened in a new browser window by simultaneously pressing the SHIFT key and clicking the link. Otherwise, the URL opens in the existing browser window when the link is clicked.

If the **Link Display Name** is blank, then the label is not underlined, and the displayed object becomes a simple text label.

The properties of the display link are:

Property	Description	Limits
Label	The link label	
Link Display Name	The name of the graphic display to be loaded when the link is clicked, or the URL of a Web page	
Label Color	The color of the label	
Label Font	The font used on the label	
Bitmap Choices	The filename of the bitmap on which to click	

Indicator Light

The indicator light displays the value of a symbol (variable) or direct address in a PLC with two possible states. An input value of 0 is considered off and a non-zero value is considered on. If the **Flash Interval** property is set to a value greater than 0, the light flashes while the input value is on. There is a bitmap for the on-state and a different one for the off-state.

The properties of the indicator light are:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 267*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 267*
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
OFF Word	The text to display when the input value is off	
OFF Bitmap Choice	The light bitmap when the OFF word is displayed	
OFF Word Color	The color of the OFF word text	
OFF Word Font	The font used for the OFF word text	
ON Word	The text to display when the input value is on	
ON Bitmap Choice	The light bitmap when the ON word is displayed	
ON Word Color	The color of the ON word font	
ON Word Font	The font used for the ON word text	
Flash Interval	The flashing time period (in ms) of the light when the input value is on. Set to 0 for no flashing.	200 to 2000
Input Inverted	If true, inverts the input value (light will show the off-word when input value is on).	
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	
PLC Value	A simulated input value for testing the graphic object	See Note 3, *Notes, p. 267*

Motor

The motor widget displays the value of a symbol (variable) or direct address in a PLC with three possible states. An input value of 0 is considered off, a value of 1 is considered on and other values are considered default. The three states are represented by different bitmaps.

The properties of the motor widget are:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 267*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 267*
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
OFF Word	The text to display when the input value is off	
OFF Bitmap Choice	The motor bitmap when the OFF word is displayed	
OFFWord Color	The color of the OFF word text	
OFF Word Font	The font used for the OFF word text	
ON Word	The text to display when the input value is ON	
ON Bitmap Choice	The motor bitmap when the ON word is displayed	
ON Word Color	The color of the ON word font	
ON Word Font	The font used for the ON word text	
DEFAULTWord	The text to display when the input value is ON	
DEFAULT Bitmap Choice	The motor bitmap when the DEFAULT word is displayed	
DEFAULT Word Color	The color of the DEFAULT word font	
DEFAULT Word Font	The font used for the DEFAULT word text	
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	
PLC Value	A simulated input value for testing the graphic object	See Note 3, *Notes, p. 267*

Pipe

The pipe displays the value of a symbol (variable) or direct address in a PLC with two possible states. An input value of 0 is considered off and a non-zero value is considered on. There is a bitmap for the on-state and a different one for the off-state.

The properties of the pipe are:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 267*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 267*
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
OFF Word	The text to display when the input value is off	
OFF Bitmap Choice	The pipe bitmap when the OFF word is displayed	
OFF Word Color	The color for the OFF word text	
OFF Word Font	The font used for the OFF word text	
ON Word	The text to display when the input value is on	
ON Bitmap Choice	The pipe bitmap when the ON word is displayed	
ON Word Color	The color of the ON word font	
ON Word Font	The font used for the ON word text	
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	
PLC Value	A simulated input value for testing the graphic object	See Note 3, *Notes, p. 267*

Push Button

A push button sends preset value(s) to a PLC when the user clicks it with the mouse. These are the properties of the Push Button.

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 267*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 267*
Background	The background color of the graphic object	
Values	The value(s) to send to the PLC	See Note 4, *Notes, p. 267*
Reset Values	The value(s) to send to the PLC after the reset delay time has expired. If no reset values are provided, no reset action will occur.	
Reset Delay	The delay time (in milliseconds) that the Push Button should use after sending the value(s) to the PLC before sending the reset value(s).	0-2000
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
Button Label	The text label for the button	
Button Label Color	The color of the button label	
Button Label Font	The font used for the button label	
OFF Bitmap Choice	The button bitmap when the OFF state is displayed	
ON Bitmap Choice	The button bitmap when the ON state is displayed	
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	

Valve

The valve displays the value of a symbol (variable) or direct address in a PLC with two possible states. An input value of 0 is considered off and a non-zero value is considered on. There is a bitmap for the on-state and a different one for the off-state.

The properties of the valve are:

Property	Description	Limits
Name	The name of the graphic object	
Address	The direct address or the name of a symbol (variable) to monitor	See Note 1, *Notes, p. 267*
Data Type	The data type of the direct address or symbol (variable)	See Note 2, *Notes, p. 267*
Background	The background color of the graphic object	
Label	The label to be displayed as part of the graphic object	
Label Color	The color of the label	
Label Font	The font used for the label	
OFF Word	The text to display when the input value is off	
OFF Bitmap Choice	The valve bitmap when the OFF word is displayed	
OFF Word Color	The color of the OFF word text	
OFF Word Font	The font used for the OFF word text	
ON Word	The text to display when the input value is ON	
ON Bitmap Choice	The valve bitmap when the ON word is displayed	
ON Word Color	The color of the ON word font	
ON Word Font	The font used for the ON word text	
Flash Interval	The flashing time period (in ms) of the light when the input value is on. Set to 0 for no flashing.	200 to 2000
Border Width	The width (in pixels) of the graphic object's border	0 to 32
Border Color	The color of the graphic object's border	
PLC Value	A simulated input value for testing the graphic object	See Note 3, *Notes, p. 267*

Notes

These are the notes for the chapter.

1. If the address property of a graphic object is a direct address, the data type property is set to **UNDEFINED**, a default data type (**BOOL**, **INT**, **DINT**, or **REAL** based on the implied size of the data value) is used. If the address property is a symbol (variable) name, the data type property does not have to be specified and can be set to **UNDEFINED**. If, however, the data type property is specified for a symbol (variable), it must exactly match the symbol's (variable's) actual data type.
If the address property is a direct address for a discreet PLC reference (Quantum 0x/1x reference), the data type property must be set to **BOOL**. The data type property may be set to **BOOL** only for a discrete PLC reference.

2. The meaning of the possible values of the Data Type property are:

Data Type	Meaning
UNDEFINED	no data type specified
BOOL	1-bit discreet (Boolean)
SHORT	8-bit signed integer
USHORT	8-bit unsigned integer
INT	16-bit signed integer
UINT	16-bit unsigned integer
DINT	32-bit signed integer
UDINT	32-bit unsigned integer
REAL	32-bit IEEE floating point
TIME	32-bit unsigned integer (in milliseconds)
DATE	Date (32-bit BCD)
TOD	Time of Day (32-bit BCD)
DT	Date and Time (64-bit BCD)

3. The limits for the Maximum PLC Value and Minimum PLC Value properties are the natural limits of the Data Type property that is set. A Data Type setting of UNDEFINED is treated as REAL with respect to its limit values.

4 For a Push Button, a minimum of one value must be provided. If the Address property is a symbol (variable) name, then only one value will ever be sent to the PLC, and any additional values will be ignored. If the Address property is a direct address, then all of the values provided will be sent to the PLC as an array, starting at the specified direct address.

8.5 Graphic Viewer

Graphic Viewer

Overview

Graphic Viewer is a lighter version of the Graphic Editor. Its small size enables you to download it faster. With Graphic Viewer you can only display widgets. You can not edit them.

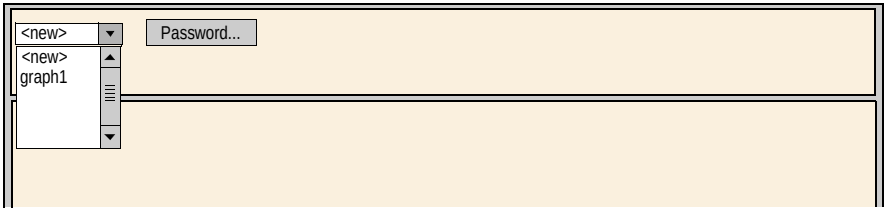
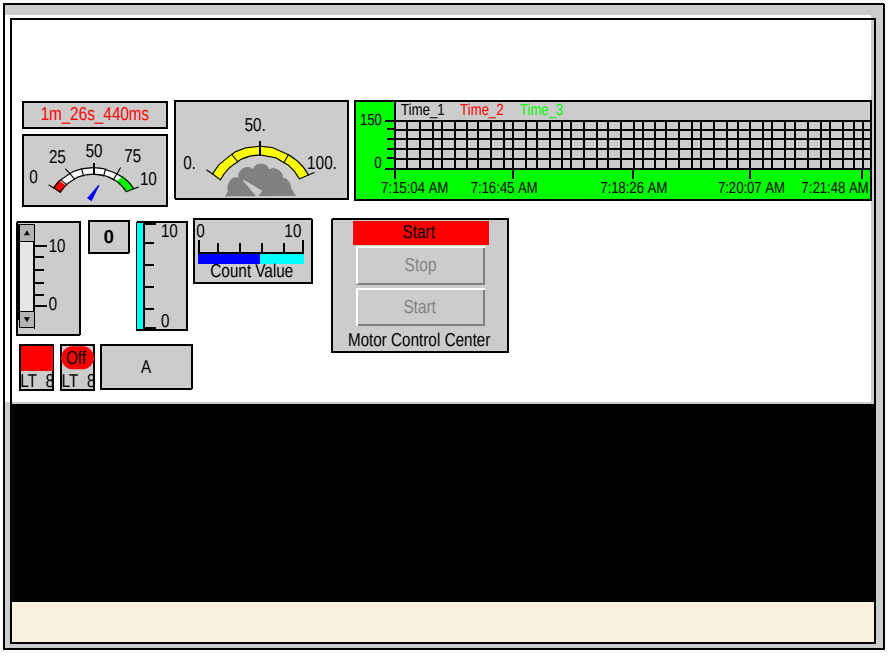
Graphic Viewer is divided into two windows:

- **Top Window**: This area shows the user controls. This is where you can select a graphic to edit from a drop-down menu,
- **Display Window**: This area displays the selected graphic.

When viewing a graphic in full-screen mode, the **Top Window** is not displayed.

Selecting a Graphic

The following instructions tell you how to select a graphic created by the Graphic Editor:

Step	Action
1	Click on **Graphic Viewer**. The **Top and Display Windows** appear: 
2	Select a graphic from the drop-down menu. **Result**: The selected graphic appears in the **Display Window**:  **Note**: Double-click in the **Display Window** area in the active graphic (unless the active graphic is a command widget) to refresh the **Top Window**.

Adding Custom Pages to the Site

9

Adding Custom Pages to the Site

Overview

You may choose to add your own Web pages to the site on the Embedded Server. The FactoryCast Configuration Tool allows you to protect these pages with the same passwords as the default pages. You can also put them in an unprotected area where anyone can view them without a password.

FactoryCast provides some simple Java applets to help you include dynamic data from the controller in your custom Web pages.

This section discusses how to add custom pages to the site and how to use the Java applets on those pages.

Note: When planning custom Web pages, be sure to keep them within the limits of the memory available for customization. For additional information, refer to Embedded Server Operations (See *Embedded Server Operations, p. 181*).

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
9.1	Working with Custom Pages	272
9.2	Using FactoryCast's Java Applets Using HTML Code	276
9.3	Using FactoryCast's Microsoft FrontPage Extension	291
9.4	Using Graphic Objects Lite	315

9.1 Working with Custom Pages

Working with Custom Pages

Overview

You may develop any pages you want in order to customize the FactoryCast site. FactoryCast Configurator gives you two different methods for adding these pages to the site:

- You may add supporting pages and give them password protection.
- You may add supporting pages and make them available to all users.

What's in this Section?

This section contains the following topics:

Topic	Page
Downloading a Custom Home Page	273
Downloading Supporting Pages	275

Downloading a Custom Home Page

Introduction

If you want to replace the default home page with one of your own, follow these steps:

Step	Action
1	Back up the original FactoryCast configuration, so that you can restore it later if needed.
2	Create your home page.
3	Place your home page in its location.
4	Download your home page to the Embedded Server.

Backing Up the Original Configuration

Before you change the default home page, you should back up the configuration (See *Backing Up the Web Server, p. 178*). That way, if problems develop with the Embedded Server, you can restore the server to its original configuration.

Creating Your Home Page

You must give your home page the same name as the default home page: index.htm.

Placing Your Home Page

Copy your home page to:

- the ‘wwwroot\secure\user\’ directory for custom pages with password protection,
- the ‘wwwroot\unsecure\user\’ directory for custom pages without password protection.

What About Image Files?

Any image files for your new home page should be placed in the same directory as your company logo.

Image filenames must comply with the DOS 8.3 naming convention. (A filename that does not exceed eight letters and its three-letter extension).

Downloading

Follow the steps in the table below to download your new home page to the Embedded Server:

Step	Action
1	Select **Transfer | Web Files** from the menu bar. **Result**: The **Confirm Host** dialog box appears with the Embedded Server's IP address and any host name. **Note:** If the IP address or host name is incorrect, make the appropriate changes in the **IP Address** dialog box. Elsewhere in this guide is information for Setting the IP Address (See *Setting the IP Address, p. 153*).
2	Click **OK**. **Result:** The **Transfer Web Files** dialog box appears.
3	Make sure that the right-hand window displays the Embedded Server's \wwwroot directory. Locate and select the new home page in the left-hand window.
4	Click the → button to transfer the home page, overwriting the index.htm file in the Embedded Server's \wwwroot directory.

Downloading Supporting Pages

Overview

If you are adding supporting Web pages to the default Web site, you may choose to protect them with the same user name and password as the default pages, or you may choose to make them accessible without a password.

In order to add pages to the site, you must specify the directory where they are located and download them to the server.

Specifying File Locations

Place the files for your Web pages and any images for the pages in a single directory. Each subdirectory should have a page named `index.htm`.

Make sure all filenames comply with the DOS 8.3 naming convention (a filename of no more than eight letters and its three-letter extension).

Use the **Settings** dialog box to specify the files' location. (Refer to *Setting File Locations, p. 154*.)

Download Options

Pages that are to be password protected must be copied to the Embedded Server's \wwwroot\secure\user directory.

Note: Remember to create new hyperlinks to the other Web pages so you can access them in the browser.

Downloading to a Server

Follow the steps below to download to the server:

Step	Action
1	Select **Transfer | Web Files** from the menu. **Result:** The **Confirm Host** dialog box appears with the Embedded Server's IP address and any host name. **Note:** If the IP address or host name is incorrect, make the appropriate changes in the **IP Address** dialog. (Refer to *Setting the IP Address, p. 153*.)
2	Click **OK** to confirm the IP address and begin the download. **Result:** The **Transfer Web Files** dialog appears.
3	Make sure that the window on the right shows the desired directory in the Embedded Server. Locate and select the new pages in the left-hand window.
4	Click the → button to transfer the pages to the Embedded Server.

9.2 Using FactoryCast's Java Applets Using HTML Code

Using FactoryCast's Java Applets on Custom Web Pages

Overview

This section describes how to use the Java applets that come with FactoryCast. Use these applets to create custom Web pages. To create custom pages, use any plain text editor, such as Microsoft Notepad.

Creating custom Web pages with HTML makes it possible to view live PLC data in your browser.

What's in this Section?

This section contains the following topics:

Topic	Page
Inserting Applets on a Web Page	277
Inserting LiveBeanApplet	278
Inserting LiveBeanApplet using HTML Code	281
Inserting LiveLabelApplet using HTML Code	284
Inserting LiveTableApplet using HTML Code	289

Inserting Applets on a Web Page

Overview

FactoryCast software includes several graphic objects (or JavaBeans) that are used to visualize live PLC data on a graphic display. Generally, an end-user builds a graphic display using the **Graphic Editor** tool that comes with FactoryCast. However, you can also build a graphic display using graphic objects inserted into a Web page via the LiveBeanApplet. Any of the graphic objects, such as an Analog Meter or a Push Button, can be inserted into a Web page so that dynamic data may be visualized outside the context of the Graphic Editor.

To view live PLC data with a browser, you may opt for one of two methods to insert a FactoryCast applet on a Web page.

- (1) Enter the HTML code found in these sections:
  - Inserting a `LiveBeanApplet` using HTML code,
  - Inserting a `LiveLabelApplet` using HTML code,
  - Inserting a `LiveTableApplet` using HTML code.
- (2) Insert a Java applet and then fill in the dialog boxes using the FactoryCast extension for Microsoft FrontPage 2000. See *Using FactoryCast's Microsoft FrontPage Extension, p. 291*.

Inserting LiveBeanApplet

Overview

This topic discusses general concepts about inserting a `LiveBeanApplet`.

Note:
Elsewhere in this guide is information for inserting a `LiveBeanApplet` using:

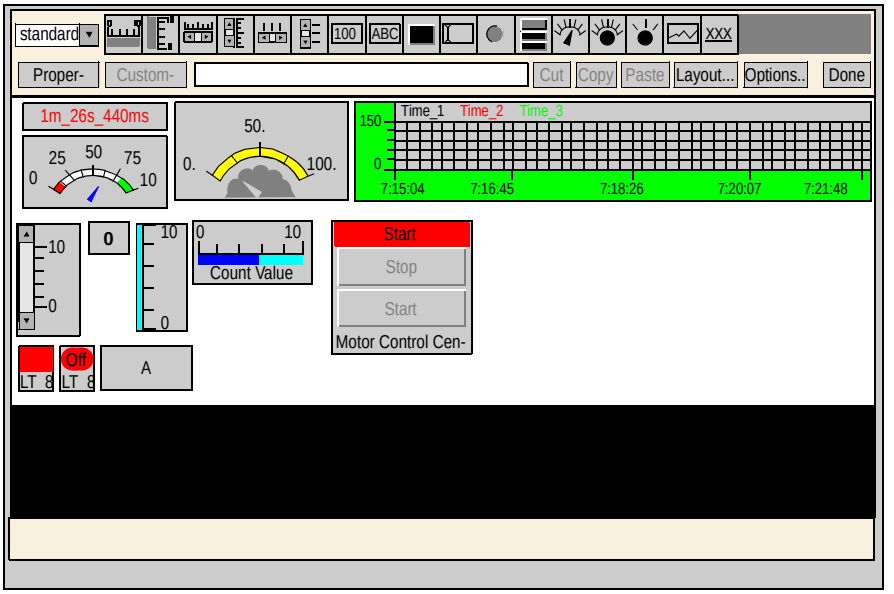
- HTML (See *Using FactoryCast's Java Applets Using HTML Code, p. 276*),
- Microsoft's FrontPage Extension (See *Installing FactoryCast's Microsoft FrontPage Extension, p. 292*).

Note: Prior to inserting a `LiveBeanApplet` into a Web page, you must create a JavaBeans library using the Graphic Editor that comes with the FactoryCast software. Generally a user will create a JavaBeans library that has one instance of every object that they would like to use in a Web page. Think of this library as a set of templates that are copied to and customized for your Web pages. For example a library may have one analog meter, one rotary selector, and one push button. Multiple instances of each bean can then be added to a Web page, each with a set of unique parameters such as an address.

Inserting a LiveBeanApplet

To insert a `LiveBeanApplet`:

Step	Action
1	Using your Web browser, enter your Web server module's name or IP address and go to the Graphic Editor applet found via the Monitoring and Online Editors Web link. 

Step	Action
2	Enter the **Edit** mode and create a JavaBeans library you want to use on your Web pages. Customize each bean with a common set of parameters so that you only have to configure unique parameters, such as the register address.  **Note:** For help using the Graphic Editor applet, see the section Graphic Editor (See *Graphic Editor, p. 220*).
3	Once you have created your library, click the **Done** button in the editor window and save your library.

Inserting LiveBeanApplet using HTML Code

Overview

To graphically visualize data, use graphic objects such as Analog Meter or Push Button. Before any beans are inserted into a Web page, the special applet called `LiveBeanMgrApplet` must be inserted into the server.

LiveBean-MgrApplet

The `LiveBeanMgrApplet` allows the Web page to display dynamic data from the controller. This applet **must** be included **once** on the page if any instances of `LiveBeanApplet` are included in the page.

The `LiveBeanMgrApplet` can be included on a Web page in two possible forms.

- Invisible applet—if the Web page is used only to monitor PLC values, then no input is needed from the user,
- Icon of a key—if the Web page is used both to send new values and to monitor values to the PLC, then input is needed from the user in order to send new values.

Note: If the applet is used as an icon of a key, the user enters a password in order to send values to the PLC. From the Web browser click on the applet (icon of a key), a dialog box appears requesting the user to enter a password; entering the password enables the user to write to the PLC.

Here is the HTML code that you use to include the applet on a Web page that is used only for monitoring:

```
<APPLET codebase="/classes"
archive="SAComm.jar,GDE.jar,Widgets.jar"
code="com.schneiderautomation.gde.LiveBeanMgrApplet"
width=0 height=0>
</APPLET>
```

Here is the HTML code that you use to include the applet on a Web page that is used for sending values to a PLC as well as monitoring:

```
<APPLET codebase="classes"
archive="SAComm.jar,GDE.jar,Widgets.jar"
code="com.schneiderautomation.gde.LiveBeanMgrApplet"
width=32 height=32>
<PARAM name=MODE value="READWRITE">
<PARAM name=AUTO_LOGIN value="FALSE">
</APPLET>
```

In the above example, if the value of the **AUTO_LOGIN** parameter is set to **TRUE**, instead of **FALSE**, then the password is automatically entered and the user is not required to enter the password.

LiveBeanApplet

The `LiveBeanApplet` is included one time for each symbol (variable) or direct address monitored/controlled on the Web page. For instance, if you are monitoring three symbols (variables), you would include the applet three times. `LiveBeanApplet` allows any graphic object/Java Bean that was created with the **Graphic Editor** to be included on a Web page as a separate applet. (See *Graphic Editor, p. 220*, for information on the **Graphic Editor**.) Any graphic object that has been saved as part of a **Graphic Editor** graphic display can be retrieved from the graphic file and presented by the applet.

LiveBeanApplet Parameters

The `LiveBeanApplet` uses parameters that allow you to specify the graphic object to be presented by the applet and to set the applet's background color.

The applet's parameters and their meanings are shown below.

Parameter...	Defines...
LIBRARY	The name of the graphic display which contains the graphic object that is to be presented by the applet. (This will be the same name that was used when the graphic display was saved with the **Graphic Editor**.) **This parameter is required.**
BEAN	The name of the graphic object that is to be retrieved from the graphic display specified by the LIBRARY parameter. (This will be the name that appears as the 'Name' property of the graphic object.) **This parameter is required.**
BACKGRND	The background color for the applet. Acceptable values are WHITE, LT_GRAY, GRAY, DK_GRAY, BLACK, RED, PINK, ORANGE, YELLOW, GREEN, MAGENTA, CYAN, and BLUE. Also, a RGB color value can be entered using the format "0xRRGGBB" where RR, GG, and BB are the hexadecimal values for the red, green, and blue components, respectively. This parameter is optional but is normally set to match the color of the HTML page.

In addition to the above parameters, the <APPLET> tag for a `LiveBeanApplet` must include **width** and **height** attributes. Normally, the size of a `LiveBeanApplet` is set to match the size of the graphic object that it is presenting. To get the size of a graphic object, select the object while the **Graphic Editor** is in editing mode. The selected object's name and size are shown in the **Information Area** at the top of the **Graphic Editor** applet.

LiveBeanApplet Example

All instances of `LiveBeanApplet` that are included in a Web page follow the same pattern, with only the applet's parameters and size varying for each instance.

Here is the HTML code for including a `LiveBeanApplet` that will present the graphic object named **MyMeter** which was saved by the **Graphic Editor** as part of the graphic display, **Library1**.

```
<APPLET codebase="/classes"
archive="SAComm.jar,GDE.jar,Widgets.jar"
code="com.schneiderautomation.gde.LiveBeanApplet"
width=180 height=160>
<PARAM name=LIBRARY value="Library1">
<PARAM name=BEAN value="MyMeter">
<PARAM name=BACKGRND value="0xDDEEFF">
</APPLET>
```

More Information

For more information about creating Java applets and graphic objects to obtain runtime data from a PLC, refer to the Software Developer's Kit included in the FactoryCast installation.

Inserting LiveLabelApplet using HTML Code

Overview

Before any live labels are inserted into a Web page, the special applet called `LiveLabelMgrApplet` must be inserted into the page.

> **Note:** However, if a Web page contains both `LiveLabelApplet` and `LiveBeanApplet`, then that page must contain a single instance of `LiveBeanMgrApplet`, not `LiveLabelMgrApplet`. (`LiveBeanMgrApplet` supports both `LiveLabelApplet` and `LiveBeanApplet`, while `LiveLabelMgrApplet` supports only `LiveLabelApplet`.)

LiveLabelMgrApplet

The `LiveLabelMgrApplet` allows the Web page to display dynamic data from the controller. This applet **must** be included **once** on the page if any instances of `LiveLabelApplet` are included on the page.

Here is the HTML code that you use to include the applet on a page.

```
<APPLET>
codebase="/classes" archive="SAComm.jar"
code="com.schneiderautomation.factorycast.LiveLabelMgrApple"
width=0 height=0>
</APPLET>
```

LiveLabelApplet

Use one `LiveLabelApplet` for every symbol (variable) or direct-address monitored on the Web page used. For example, if you are monitoring three symbols (variables), you would include the applet three times.

This applet displays the following three fields.

Field	Description
Label	Your label for the symbol (variable) or direct address
Value	Run-time value of the symbol (variable) or direct address
Units	The units you specify for the value

Data Parameters The applet's parameters, their meaning, and the default values are shown below.

Parameter...	Defines...		With Default Value of...
LABEL	A text label to identify the data item		No label
UNITS	A text label to identify the value's engineering units		No units displayed
ADDRESS	The name of Concept/PL7/Unity Pro symbol (variable) or Quantum/Premium/Micro direct address		None
DATATYPE	The data type of the symbol (variable) or direct address. Acceptable values for this parameter are		UNDEFINED
	SHORT	8-bit signed integer	
	USHORT	8-bit unsigned integer	
	INT	16-bit signed integer	
	UINT	16-bit unsigned integer	
	DINT	32-bit signed integer	
	UDINT	32-bit unsigned integer	
	REAL	32-bit IEEE floating point	
	TIME	32-bit unsigned integer (in ms)	
	DATE	Date (32-bit BCD)	
	TOD	Time of Day (32-bit BCD)	
	DT	Date and Time 64-bit BCD	
	BOOL	1-bit discrete (boolean)	
	NOTES: If the ADDRESS parameter is a direct address, and the DATATYPE parameter is not specified, a default DATATYPE (BOOL ,INT,DINT or REAL based on the implied size of the data value) is used. If ADDRESS is a direct address for a discrete PLC reference (Quantum 0x/1x reference), DATATYPE must be set to BOOL. DATATYPE may be set to BOOL only for discrete PLC references. If the ADDRESS parameter is the name of a Concept, PL7 or Unity Pro symbol (variable), the DATATYPE parameter is optional. If the DATATYPE is specified for a symbol (variable), it must exactly match its actual data type,TIME is not a valid data type for PL7 Premium.		

Parameter...	Defines...		With Default Value of...
FORMAT	The display format for the value. Acceptable values for this parameter are		DEC for most data types TIME for data type TIME BOOL for data type BOOL DATE for data types DATE, TOD and DT.
	DEC	decimal	
	HEX	hexadecimal	
	BIN	binary	
	ASCII	bytes displayed as ASCII characters	
	TIME	'day_hr_min_sec_ms'	
	DATE	'yyyy-mm-dd-hh and /or hh:mm:ss	
	BOOL	ON_WORD or OFF_WORD (see below)	
	Note: If DATA TYPE is REAL, a FORMAT other than DEC will give unpredictable results if the value cannot be converted to an integer.		
GAIN	The gain (multiplier) used for scaling the retrieved value to engineering units.		1.0
	Note: Scaling is to be performed only if GAIN or BIAS is set and FORMAT is DEC. Linear scaling is performed by the formula: SCALED_VALUE=GAINxRAW_VALUE+BIAS		
BIAS	The bias (offset) used for scaling the retrieved value to engineering units. See NOTE for GAIN.		0.0
ON_WORD	A text value to be shown when value is non-zero (Use only if the FORMAT is BOOL).		ON
OFF_WORD	A text value to be shown when value is zero (Use only if the FORMAT is BOOL).		OFF
FOREGRND	Foreground color of the applet. Acceptable values are: WHITE, LT_GRAY, DK_GRAY, BLACK, RED, PINK, ORANGE, YELLOW, GREEN, MAGENTA, CYAN, and BLUE Also, a RGB color value can be entered using the format "0xRRGGBB" where RR, GG, and BB are the hexadecimal values for the red, green, and blue components, respectively.		BLACK
BACKGRND	Background color for the applet. For acceptable values, see FOREGRND.		LT_GRAY

Parameter...	Defines...	With Default Value of...
ERROR_ COLOR	Foreground color of the VALUE field when unable to retrieve the value from the PLC. For acceptable values, see FOREGRND.	MAGENTA
LABEL_ ALIGN	Alignment of the text in the LABEL field, if the width of the field is greater than the length of the text. Acceptable values are: LEFT, CENTER, and RIGHT.	LEFT
VALUE_ ALIGN	Alignment of the text in the VALUE field, if the width of the field is greater than the length of the text. Acceptable values are: LEFT, CENTER, and RIGHT.	LEFT
UNITS_ ALIGN	Alignment of the text in the UNITS field, if the width of the field is greater than the length of the text. Acceptable values are: LEFT, CENTER, and RIGHT.	LEFT
FONT_ NAME	Name of the font used by the applet. Acceptable values are: SERIF, SANSSERIF, and MONOSPACE.	SANSSERIF
FONT_ BOLD	If set, displays all text in the applet as bold. Acceptable values are: TRUE and FALSE.	FALSE
FONT_ ITALIC	If set, displays all text in the applet in italics. Acceptable values are: TRUE and FALSE.	FALSE
FONT_SIZE	Sets the point size of the font used by the applet.	12
LABEL_ WIDTH	The width of the LABEL field.	
UNITS_ WIDTH	The width of the UNITS field.	

Size Parameters

The size of a `LiveLabelApplet` is specified in the width and height attributes of its <APPLET> tag. Unless the width of Label or Units field is set with the LABEL_WIDTH or UNITS_WIDTH parameters, the LABEL and UNITS fields of the applet will always take the width required to display the text values of their associated applet parameters. The remaining width of the applet is given to its VALUE field.

LiveLabelapplet Example #1

The applet example in this section contains almost every applet parameter. Here is the HTML code for this example.

```
<APPLET codebase="/classes" archive="SAComm.jar"
code="com.schneiderautomation.factorycast.LiveLabelApplet"
width=300 height=30>
<PARAM name=LABEL value="Reactor 1 Temperature">
<PARAM name=UNITS value="F">
<PARAM name=ADDRESS value="40101">(ForPremium value="%MW100")
<PARAM name=DATATYPE value="UINT">
<PARAM name=FORMAT value="DEC">
<PARAM name=GAIN value="2.0">
<PARAM name=BIAS value="100.0">
<PARAM name=FOREGRND value="WHITE">
<PARAM name=BACKGRND value="BLACK">
<PARAM name=ERROR_COLOR value="RED">
<PARAM name=FONT_NAME value="SERIF">
<PARAM name=FONT_BOLD value="TRUE">
<PARAM name=FONT_ITALIC value="FALSE">
<PARAM name=FONT_SIZE value="10">
</APPLET>
```

LiveLabelApplet Example #2

This is an example of a minimal applet, using default values for most parameters. Here is the HTML code for this example.

```
<APPLET codebase="/classes" archive="SAComm.jar"
code="com.schneiderautomation.factorycast.LiveLabelApplet"
width=300 height=30>
<PARAM name=LABEL value="Reactor 1 Pressure">
<PARAM name=UNITS value="PSI">
<PARAM name=ADDRESS value="PT_101">
</APPLET>
```

Inserting LiveTableApplet using HTML Code

Overview

`LiveTableApplet` is used to display dynamic, runtime PLC data on a Web page, and `LiveTableApplet` operates in a manner similar to `LiveLabelApplet`. However, there is a difference: `LiveTableApplet` can show multiple input values using a tabular format. Only a single input value can be shown with `LiveLabelApplet`.

LiveTableApplet

LiveTableApplet supports applet parameters for

- Setting the number of rows in its table (N_ROWS)
- Default settings for a row's properties
- Unique settings for each row's properties

The properties (default or unique) that can be set for a row of `LiveTableApplet` are the same properties that can be set for `LiveLabelApplet`. (The parameter names are the same as those for `LiveLabelApplet`, except that **Rx**_ prefixes them, where **x** equals the applicable row number. A default setting is specified by not including the **Rx**_ prefix.

LiveTableApplet Example

This is an example of `LiveTableApplet` that has two rows and an overall width of 200. The common row properties set the width of the **Label** field to 100 and the width of the **Units** field to 40 (leaving a width of 60 for the **Value** field). Also, every row has a black background with white text of size-10 font. The **Label** text is centered; the **Value** text is right-aligned; and the **Units** text is left-aligned. The address, data type, and the text for the **Label** and **Units** fields are set individually for each of the two rows.

Here is the HTML code for this example.

```
<APPLET codebase="/classes" archive="SAComm.jar"
code="com.schneiderautomation.factorycast.LiveTableApplet"
width="200" height="40">
<PARAM name=N_ROWS value="2">
<PARAM name=LABEL_WIDTH value="100">
<PARAM name=UNITS_WIDTH value="40">
<PARAM name=BACKGRND value="BLACK">
<PARAM name=FOREGRND value="WHITE">
<PARAM name=FONT_SIZE value="10">
<PARAM name=LABEL_ALIGN value="CENTER">
<PARAM name=VALUE_ALIGN value="RIGHT">
<PARAM name=UNITS_ALIGN value="LEFT">
<PARAM name=R1_LABEL value="Reactor Pressure">
<PARAM name=R1_UNITS value="PSIG">
<PARAM name=R1_ADDRESS value="400101">
» (for Premium value="%MW101")
<PARAM name=R1_DATATYPE value="INT">
<PARAM name=R2_LABEL value="Reactor Temperature">
<PARAM name=R2_UNITS value="F">
<PARAM name=R2_ADDRESS value="400102">
» (for Premium value="%MW102")
<PARAM name=R2_DATATYPE value="INT">
</APPLET>
```

9.3 Using FactoryCast's Microsoft FrontPage Extension

Using FactoryCast's Microsoft FrontPage Extension

Overview

The *Using Factory Cast's Java Applets* section described how to add FactoryCast's Java applets to a HTML document using any text editor. This section describes using an extension for Microsoft's FrontPage application. The extension allows a user of FrontPage to easily insert FactoryCast applets to view real-time PLC data on a Web page.

What's in this Section?

This section contains the following topics:

Topic	Page
Installing FactoryCast's Microsoft FrontPage Extension	292
Inserting LiveBeanApplet Using FrontPage	298
Inserting LiveLabelApplet Using FrontPage	307
Inserting LiveTableApplet Using FrontPage	311

Installing FactoryCast's Microsoft FrontPage Extension

Overview

This section describes how to install/remove the FactoryCast Extension for Microsoft FrontPage 2000.

Installing FactoryCast

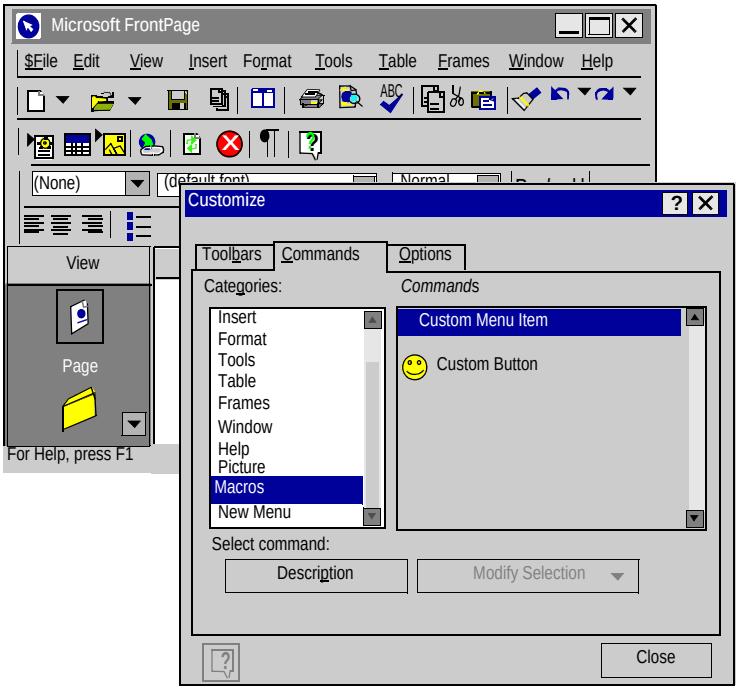
During the installation of FactoryCast, if FrontPage 2000 has been installed on the same PC, then the FactoryCast Extension for FrontPage 2000 is automatically installed as a FrontPage "Macro File." If this is the case, then proceed to the next section, which provides instructions for adding the Extension to FrontPage's menu.

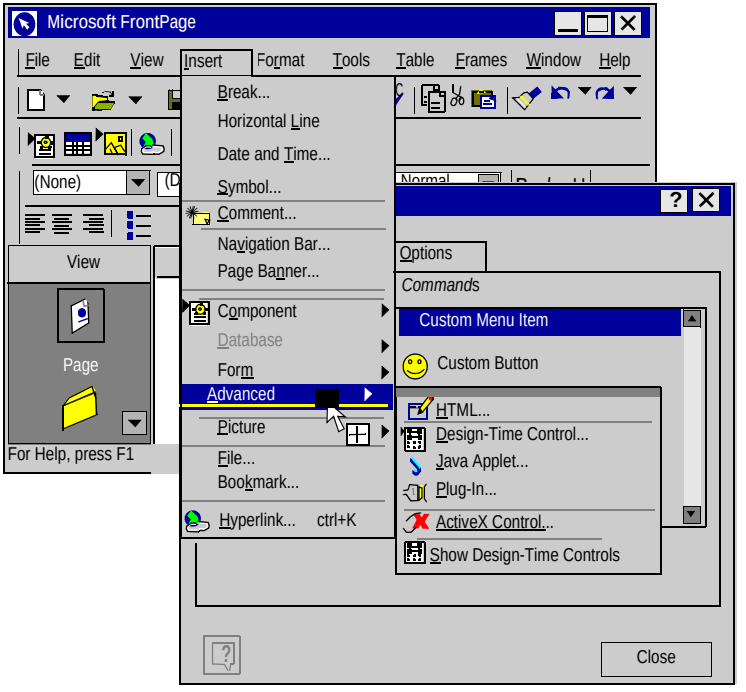
However, if FrontPage 2000 is installed after FactoryCast has been installed, then the Extension can be installed by either re-installing FactoryCast or manually copying the macro file to FrontPage's macro folder. For manual installation, after installing FrontPage 2000, the following will install the Extension:

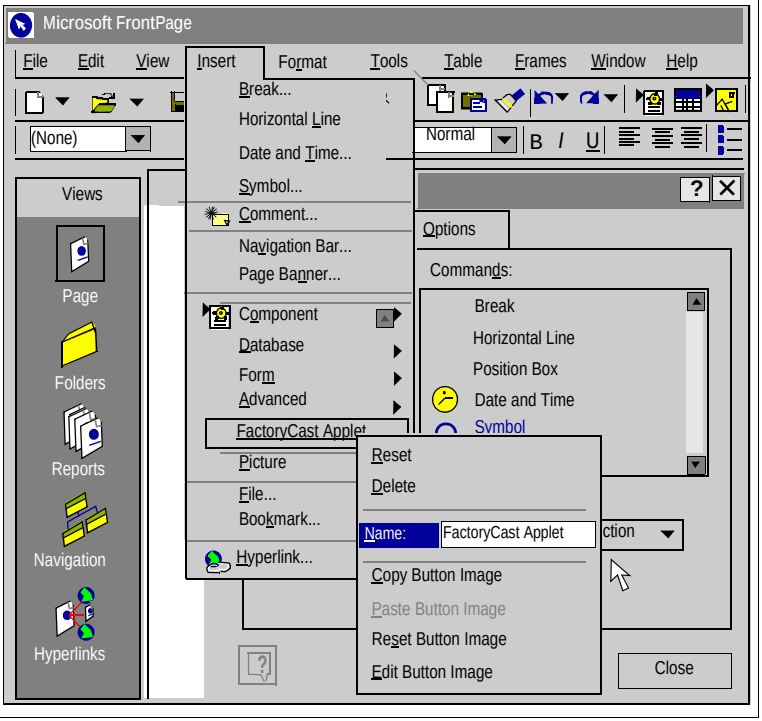
- **For Windows 95/98:** Copy the file "Microsoft FrontPage.fpm" to the folder C:\WINDOWS\Application Data\Microsoft\FrontPage\Macros (create the final "Macros" folder, if it does not already exist).
- **For Windows NT/2000/XP:** Copy the file "Microsoft FrontPage.fpm" to the folder %USERPROFILE%\Application Data\Microsoft\FrontPage\Macros (create the final 'Macros' folder, if it does not already exist). The value of the USERPROFILE environment variable is usually C:\WINNT\Profiles\<username> for NT, or C:\Documents and Settings\<username> for Windows XP. Its value can be determined by typing SET from a Command Prompt.

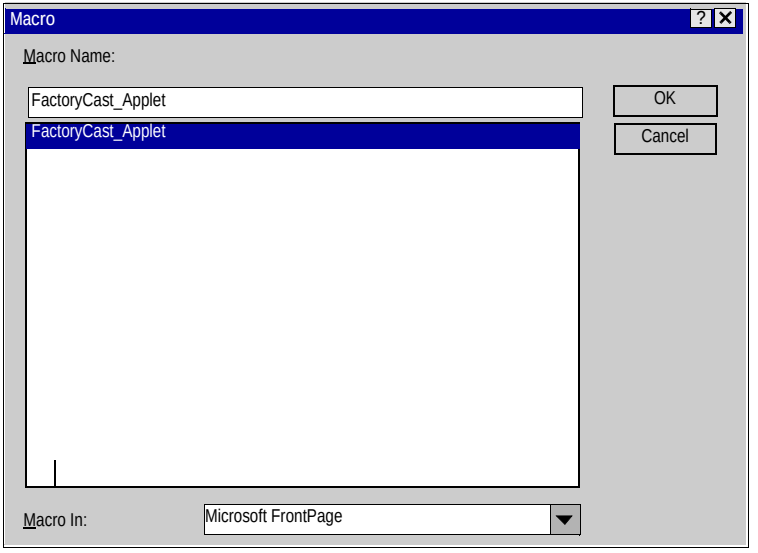
Adding FactoryCast Extension

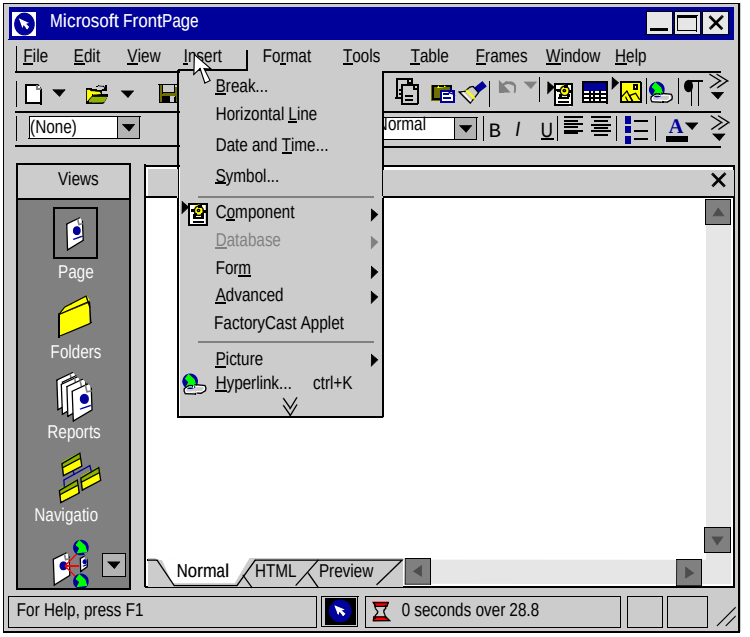
To add the FactoryCast Extension to the FrontPage **Insert** menu, do the following.

Step	Action
1	Start FrontPage 2000.
2	Click **Customize** on the **Tools** menu.
3	Click the **Commands** tab, and then select **Macros** from the **Categories** list. 

Step	Action
4	Drag and drop the entry **Custom Menu Item** from the **Commands** list to the **Insert** menu, beneath the **Advanced** command. (The **Insert** menu will automatically drop down when you drag over it.) 

Step	Action
5	Click **Modify Selection**, change the name to **FactoryCast Applet**, and then press ENTER key. 

Step	Action
6	Click **Modify Selection** again, and then choose **Assign Macro**. 
7	Choose **FactoryCast_Applet** from the list, and then click **OK**.

Step	Action
8	Click **Close**. The **FactoryCast Applet** command is added to the **Insert** menu. 

Removing FactoryCast Extension

To remove the FactoryCast Extension from the FrontPage menu, do the following.

Step	Action
1	In FrontPage, click **Customize** on the **Tools** menu.
2	Click the **Insert** menu, and then select **FactoryCast Applet**.
3	Right click, and then select **Delete** from the popup menu.

Editing Applets

There are two ways to edit an applet that has been inserted into your Web page. First, you can double-click on the object and make changes via dialog boxes. Or you can switch to the HTML editor in FrontPage and do your editing in this environment. It is suggested you edit via the first approach unless you are comfortable programming in the HTML language used to build Web pages.

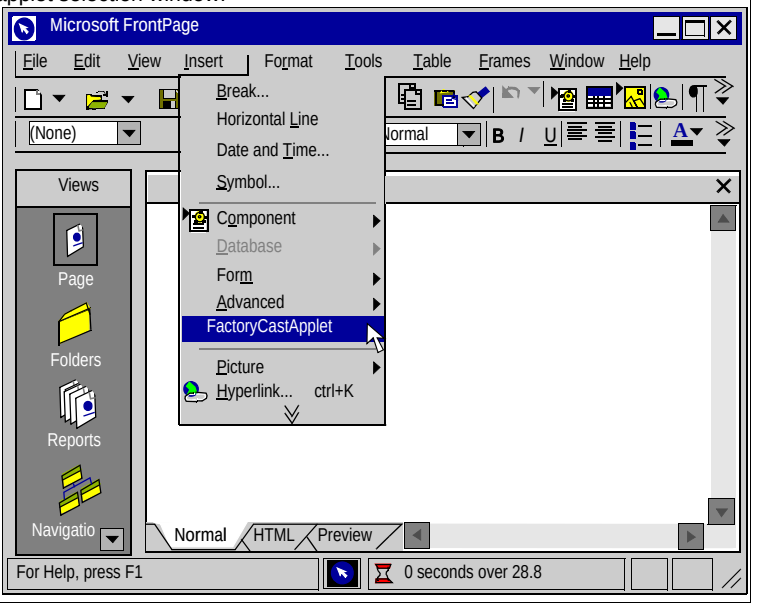
Inserting LiveBeanApplet Using FrontPage

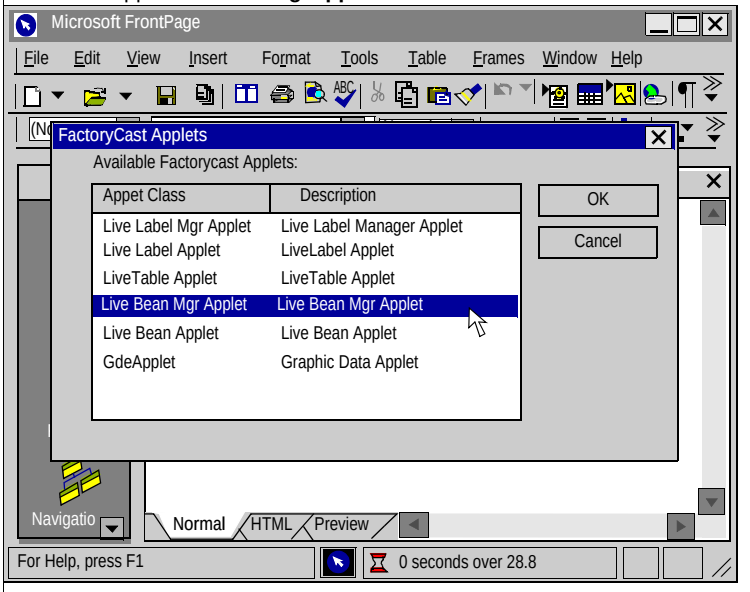
Overview

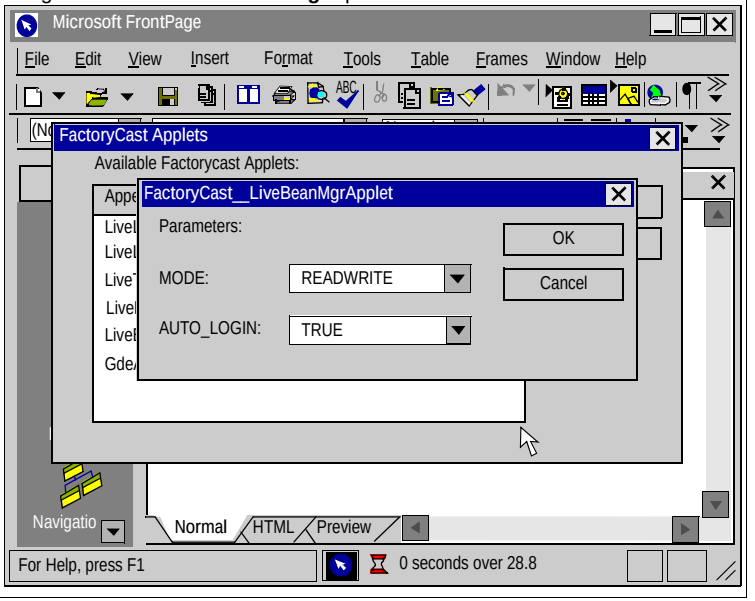
This section describes inserting `LiveBeanApplet` into a Web page. Before inserting, please read the following note and then follow steps one through sixteen below.

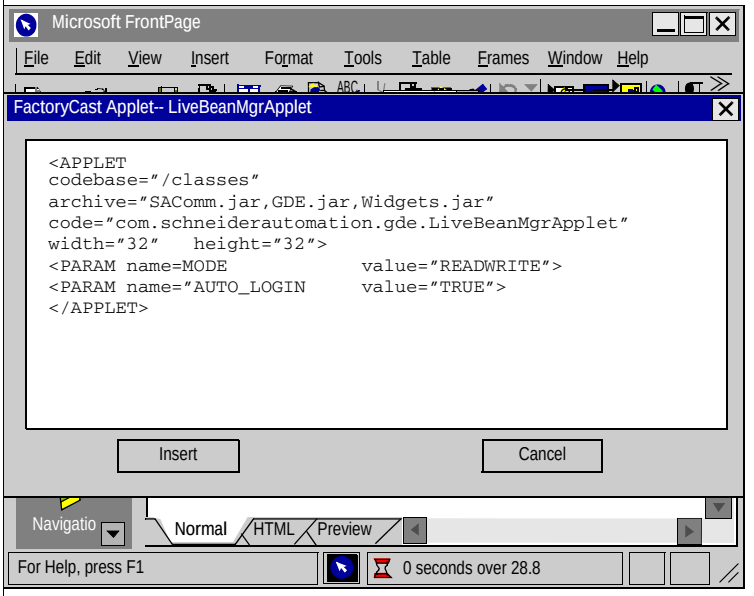
Inserting a LiveBeanApplet

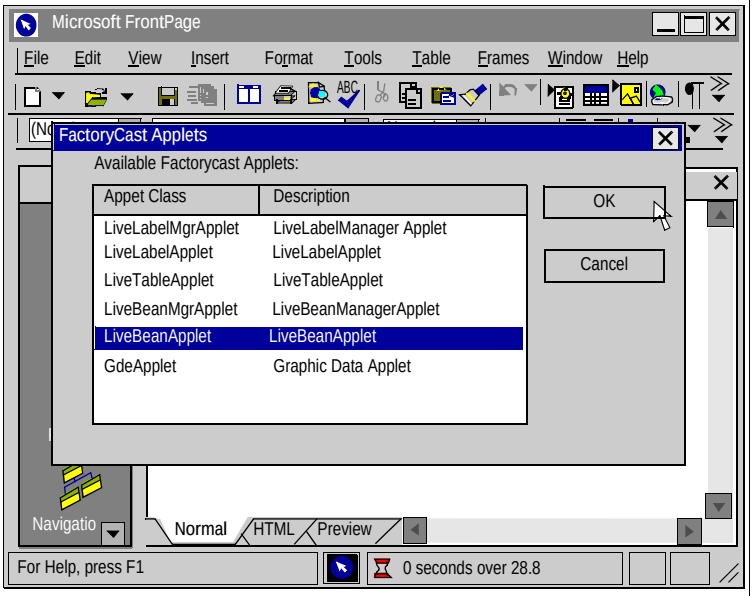
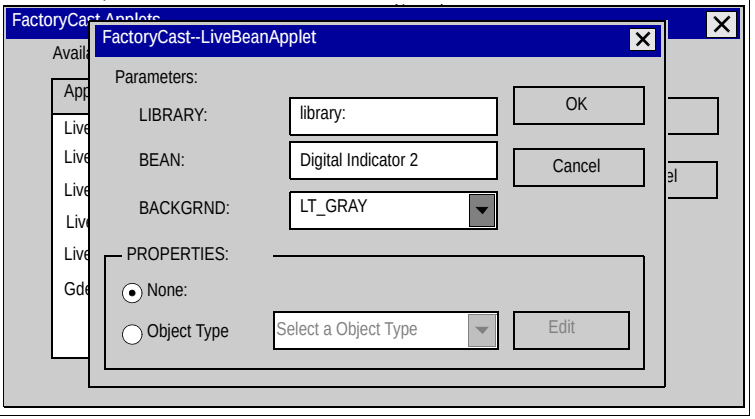
To insert a LiveBeanApplet, do the following.

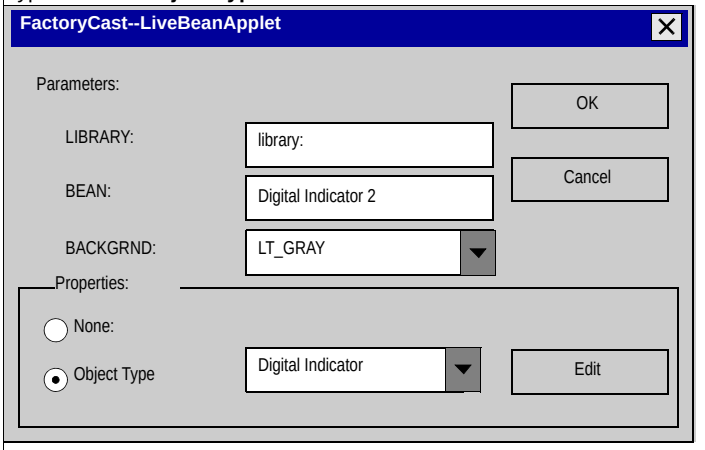
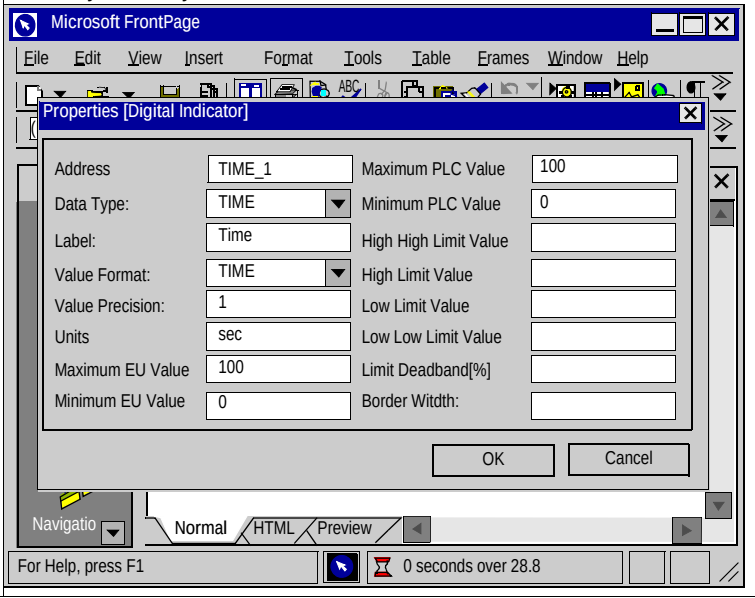
Step	Action
1	In FrontPage, Select **Insert | FactoryCast Applet** to view the FactoryCast applet selection window. 

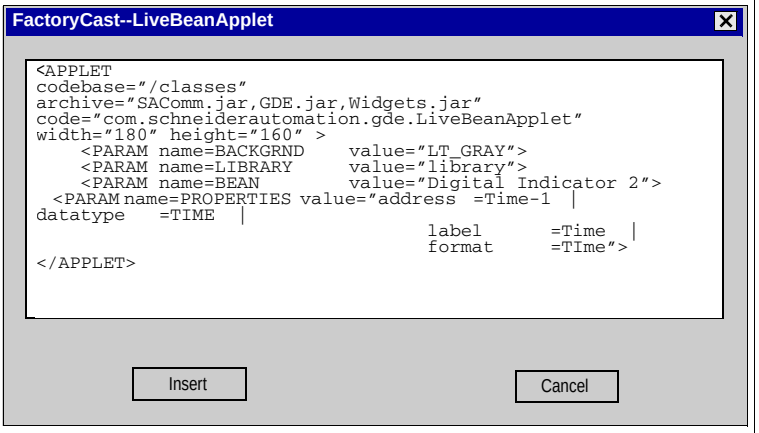
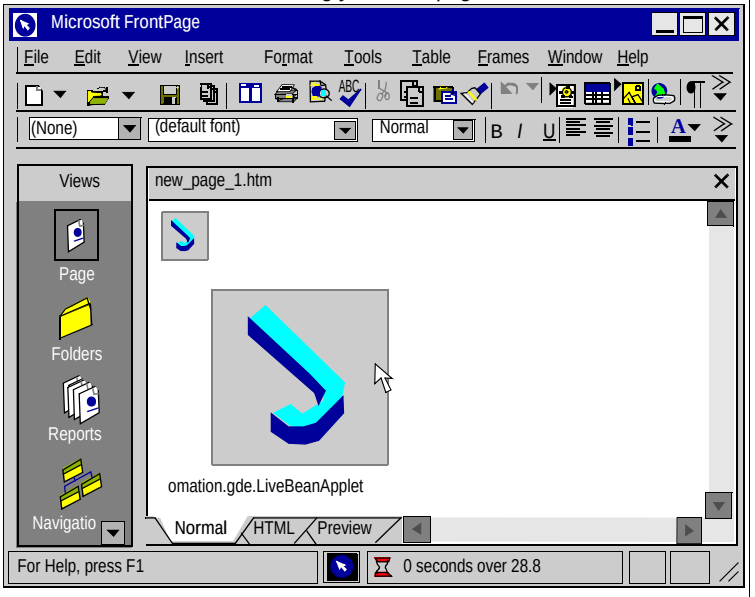
Step	Action
2	**Note:** The **LiveBeanMgrApplet** is a special applet that is inserted one time only into your Web page. Select the applet **LiveBeanMgrApplet** and then click the OK button. 

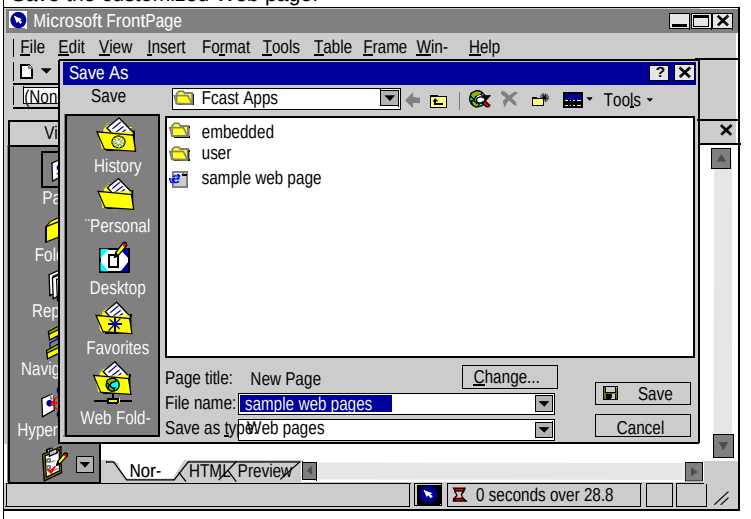
Step	Action
3	Configure the **Mode** and **Auto Login** parameters and then click the **OK** button. 

Step	Action
4	A window will appear showing the Java code that will be inserted into your HTML document.  Select the Insert button to finish inserting this applet. **Note:** Though a gray box with a blue letter "J" will appear on your Web page during editing, this applet will only be visible on your loaded Web page if you configured the Mode parameter as **ReadWrite** (it will appear as a Key when viewed via your browser). Otherwise, if **Mode** is set as **ReadOnly** then the width and height of the applet will be set to zero and you will not see it in your browser. For more information on the **Mode** parameter and its functionality, see *Using FactoryCast's Java Applets Using HTML Code, p. 276*.

Step	Action
5	Select the **LiveBeanApplet** from the FactoryCast applet selection window and then click the **OK** button. 
6	Enter the library name and the name of a particular bean in this library that you want to show. If none of the bean's properties, such as address, need to be customized, then click the **OK** button. 

Step	Action
7	Usually at a minimum the address property will need to be customized for your beans. Thus after entering the library name and bean name, select the object type from the **Object Type** box.  Click the **Edit** button when finished.
8	In the properties edit window, change only those parameters, such as address, that are unique to this bean. All other parameters will be set to that of the saved bean in your library. Click the **OK** button when finished. 

Step	Action
9	Click the **OK** button. A window will appear showing the Java code that will be inserted into your HTML document. Select the **Insert** button to finish inserting this bean. 
10	Continue to add instances of `LiveBeanApplet` to your Web page. Once you have inserted your last applet, click the **Cancel** button on the FactoryCast applet selection window to return to editing your Web page. 

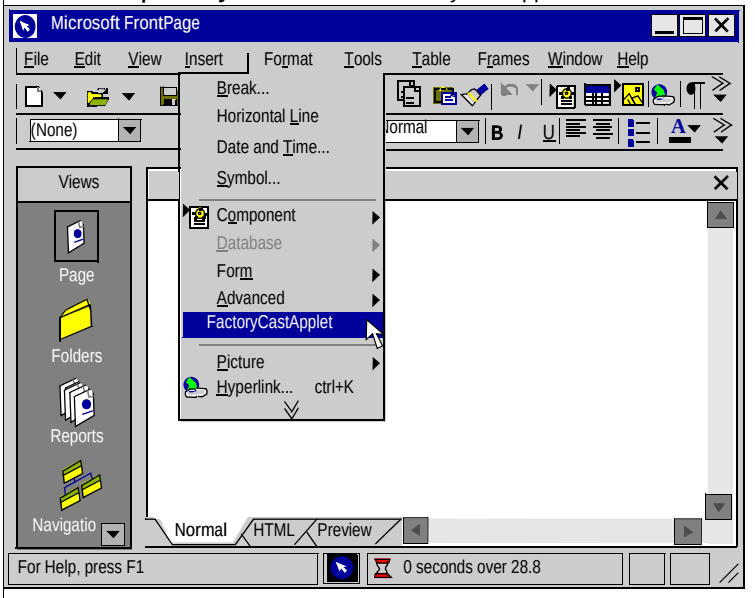
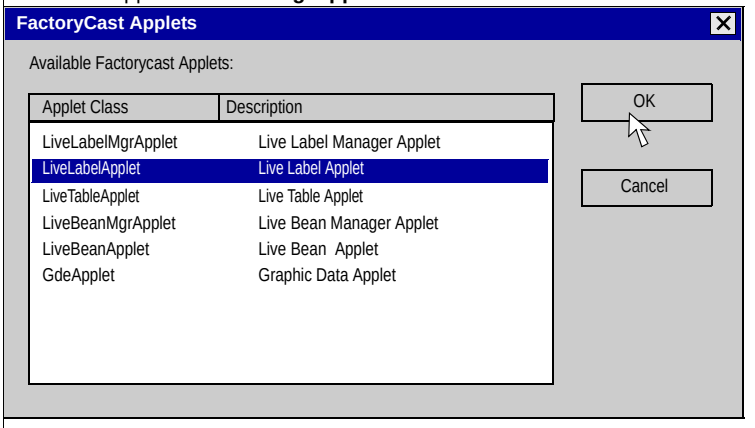
Step	Action
11	Save the customized Web page. 
12	Transfer the customized Web page to the NOE Web server.
13	Test the application.

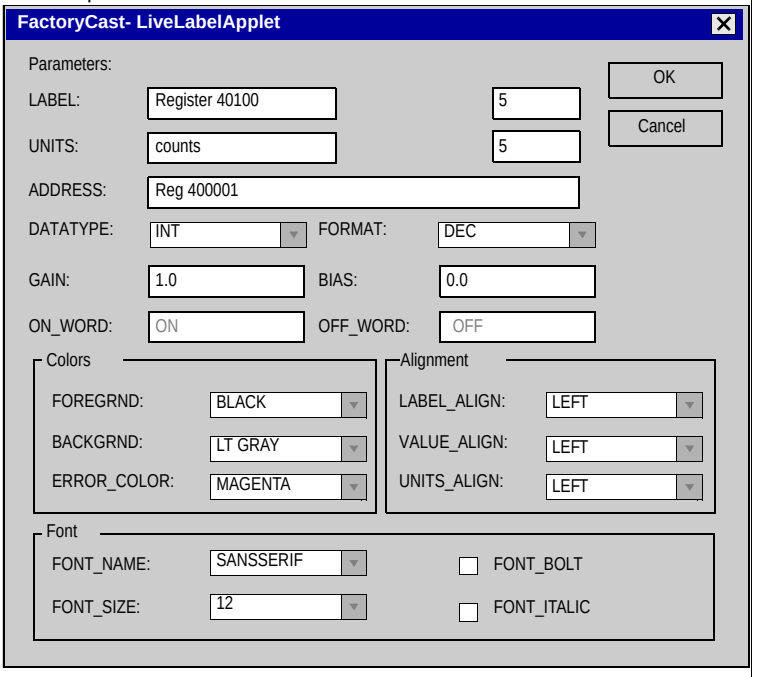
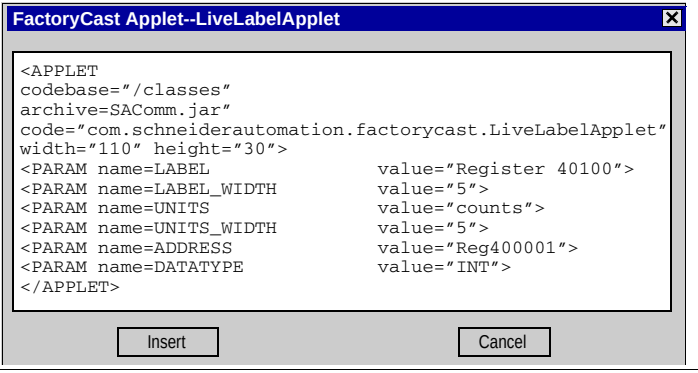
Inserting LiveLabelApplet Using FrontPage

Overview This section describes inserting LiveLabelApplet into a Web page.

Inserting LiveLabelApplet

To insert a LiveLabelApplet, do the following.

Step	Action
1	Select **Insert | FactoryCast** to view the FactoryCast applet selection window. 
2	Select the applet **LiveLabelMgrApplet** and then click the **OK** button.  **Note:** The **LiveLabelMgrApplet** is a special applet that is inserted one time only into your Web page. Though a gray box with a blue letter "J" will appear on your Web page during editing, this applet will not be visible when you view your Web page from the Web server module.

Step	Action
3	Select the applet **LiveLabelApplet** and then click the **OK** button. The Parameters edit window appears.
4	Fill in the parameters for the label and then click the **OK** button.  **Note:** For a description on each parameter, see *Using FactoryCast's Java Applets Using HTML Code, p. 276*.
5	A window will appear showing the HTML code that will be inserted into your HTML document. Click the **Insert** button to finish inserting this applet. 

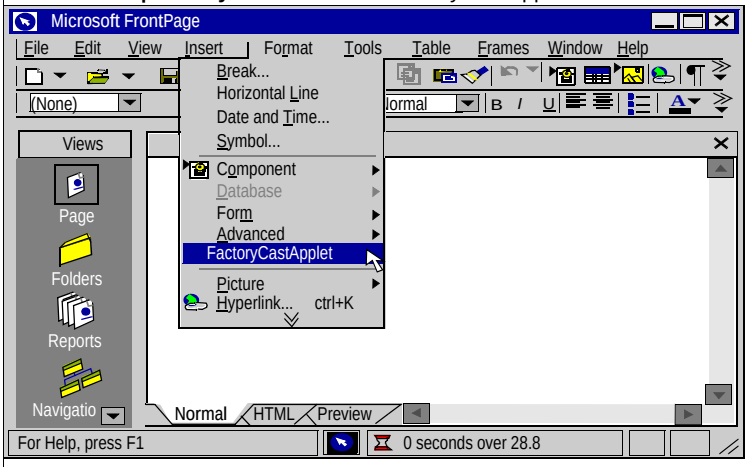
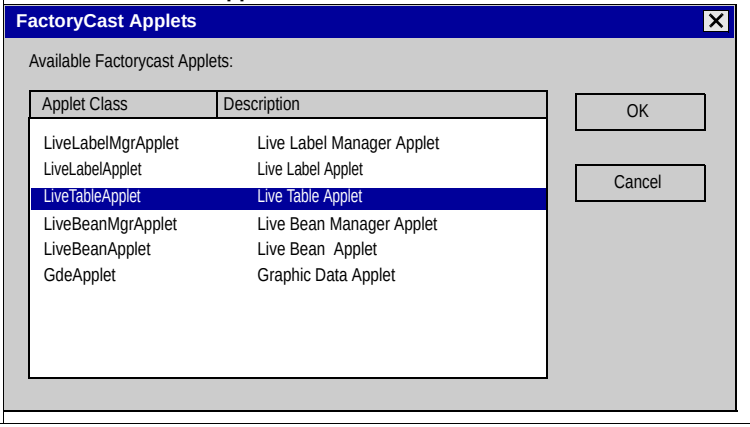
Step	Action
6	Continue to add additional instances of `LiveLabelApplet` to your Web page. Once you have inserted your last applet, click the **Cancel** button on the FactoryCast applet selection window to return to editing your Web page.

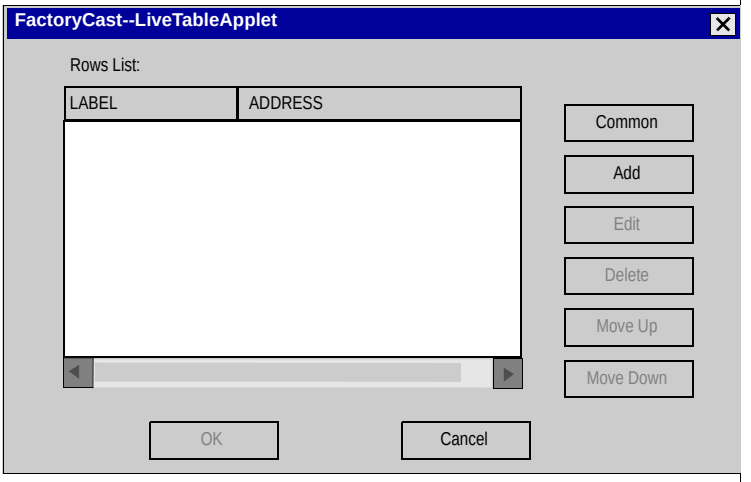
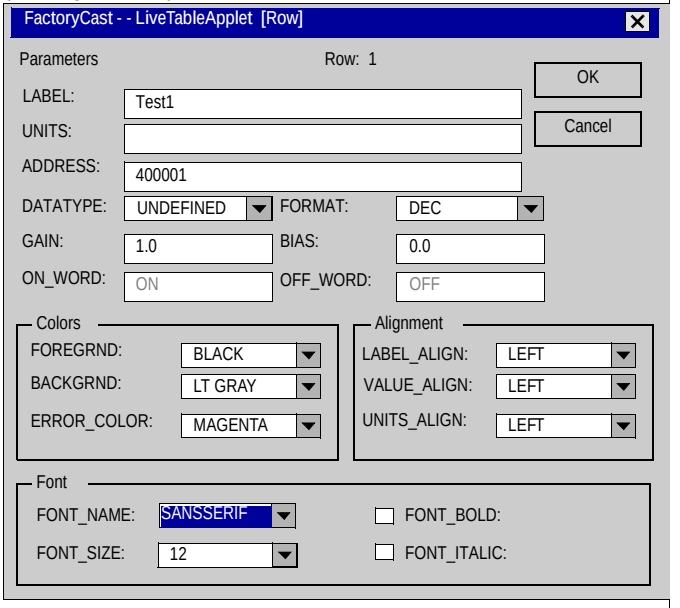
Inserting LiveTableApplet Using FrontPage

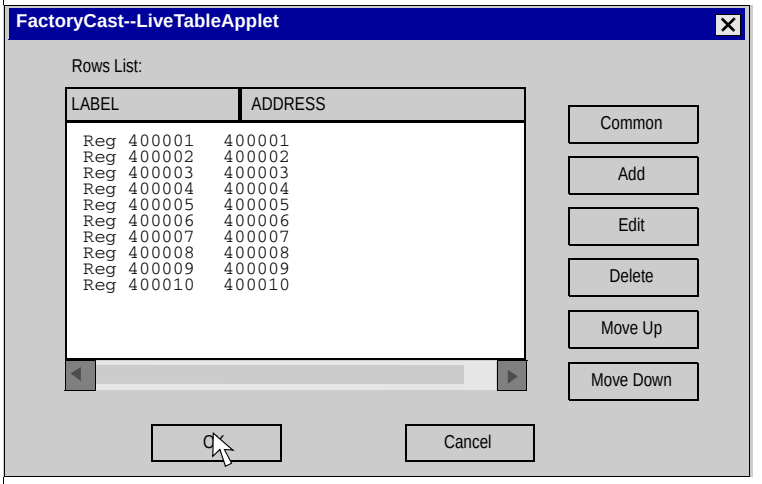
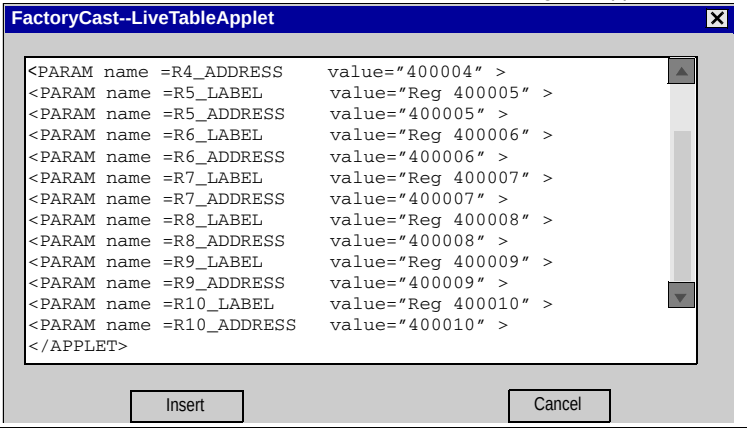
Overview This section describes inserting LiveTableApplet into a Web page.

Inserting LiveTableApplet

To insert a LiveTableApplet, do the following.

Step	Action
1	Select **Insert | FactoryCast** to view the FactoryCast applet selection window. 
2	Select the applet **LiveLabelMgrApplet** and then click the **OK** button. **Note:** The **LiveLabelMgrApplet** is a special applet that is inserted one time only into your Web page.
3	Select the **LiveTableApplet** and then click the **OK** button. 

Step	Action
4	Click the **Common** button to configure all the parameters that are common to all rows of the table.  This procedure speeds configuration by simultaneously entering many values that have the same parameter fields. You many still edit parameters for rows individually during the add process. 

Step	Action
5	Click the **Add** button to add a row to your table. The parameters edit dialog window appears. Configure parameters and then click the **Add** button when you are done. 
6	Continue to add rows using the **Add** button. You may also select a row and move it up or down the rows of your table using the **Move Up** and **Move Down** buttons. When you are finished adding and editing rows in your table, click the **OK** button from the **Rows List** window.
7	A window will appear showing the HTML code that will be inserted into your HTML document. Click the **Insert** button to finish inserting this applet. 

9.4 Using Graphic Objects Lite

Using Graphic Objects Lite

Overview

This section describes how to use the Graphic Objects Lite library. This library uses a modem and allows faster download of the graphical interface by the user. This is a lighter version of the standard graphical library.

What's in this Section?

This section contains the following topics:

Topic	Page
Downloading the Graphic Objects Lite Library	316
Description of Graphic Objects Lite	317

Downloading the Graphic Objects Lite Library

Overview

You must download the Graphic Objects Lite library before you use it. Use the Transfer Web File utility (See *Custom Web File, p. 173*) provided with the FactoryCast Configurator, and transfer the file **widgetslite.jar** to the module's Web site.

Note: The **widgetslite.jar** file is located in the installation folder, in the subdirectory **/addons/jar**.

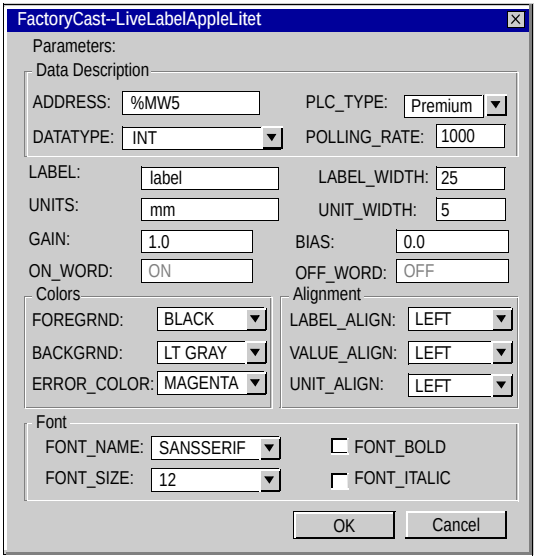
Description of Graphic Objects Lite

Overview

The set of graphic objects provided in the FactoryCast_Applet can help you create graphic displays similar to the human-machine interface (HMI) screens. All the data control and monitoring objects have integrated communication functions and are designed as standalone graphic objects.

LiveLabel AppletLite Setup

The **LiveLabel AppletLite** window displays the direct address value of a Modbus slave in a text field.



The properties of this widget are as follows:

Property	Description	Limits
Address	Address of the PLC variable	Note 1 (See *Notes, p. 335*)
Data type	PLC address data type	Note 2 (See *Notes, p. 335*)
PLC Type	Type of PLC	Premium or Quantum
Label	Label displayed as part of the graphic object	Note 5 (See *Notes, p. 335*)
Label Width	Width of label	
Unit Width	Width of unit	
Polling rate [ms]	Scanning value	
Gain	The gain (multiplier) is used for the scale of the value retrieved in physical units	1.0
Bias	The bias (multiplier) is used for the scale of the value retrieved in physical units	0.0
ON_Word	Text value displayed when the value is not zero (use if the data type is binary)	ON

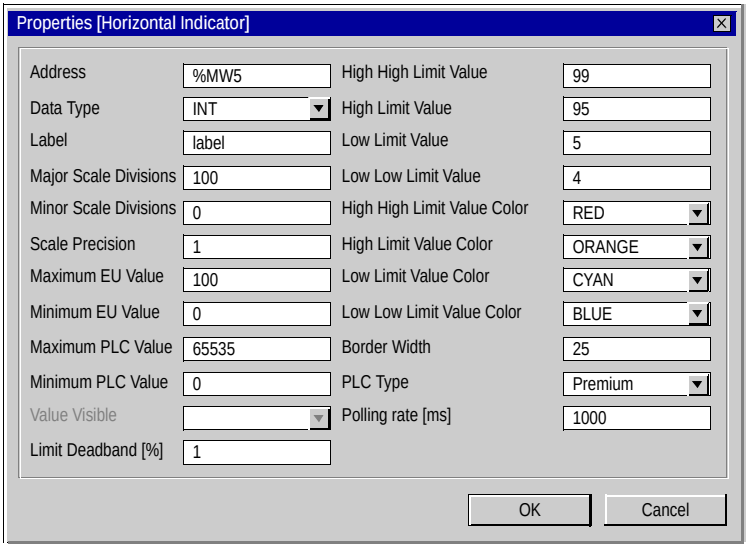
Property	Description	Limits
OFF_Word	Text value displayed when the value is not zero (use if the data type is binary)	OFF
Foregrnd	Color of the applet foreground	BLACK
Backgrnd	Color of the applet background	LT_GRAY
Error_Color	Color of the applet foreground if the address value cannot be retrieved	MAGENTA
Label_Align	Alignment of text in the Label field if the size is greater than the text length	LEFT
Value_Align	Alignment of text in the Value field if the size is greater than the text length	LEFT
Units_Align	Alignment of text in the Units field if the size is greater than the text length	LEFT
Font_Name	Font name for the applet text	SANSSERIF
Font_Bold	Applet text is bold if configured as TRUE	FALSE
Font_Italic	Applet text is italic if configured as TRUE	FALSE
Font_Size	Applet text size	12

The HTML code with the parameters of the widget above is as follows:

```
<APPLET
codebase="/classes"
archive="mbclient.jar,widgetslite.jar"
code="com.schneiderautomation.factorycast.gateway.widgetsl
ite.LiveLabelApplet"
width="130" height="30">
   <PARAM name="progressbar"    value="true">
   <PARAM name="progresscolor"    value="#000000">
   <PARAM name="ADDRESS"    value="1">
   <PARAM name="UNITID"    value="0">
   <PARAM name="RATE"    value="1000">
   <PARAM name="DATATYPE"    value="REGISTER">
   <PARAM name="LABEL"    value="label">
   <PARAM name="LABEL_WIDTH"    value="25">
   <PARAM name="UNITS"    value="mm">
   <PARAM name="UNITS_WIDTH"    value="5">
</APPLET>
```

Horizontal or Vertical Indicator Setup

The horizontal or vertical indicator gives an analog representation of the value of a direct address of a Modbus slave. It is a horizontal or vertical bar whose length is proportional to the value. It represents a percentage of its range in physical units.



The properties of the indicator are as follows:

Property	Description	Limits
Address	Address of the PLC variable	Note 1 (See *Notes, p. 335*)
Data type	PLC address data type	Note 2 (See *Notes, p. 335*)
Label	Label displayed as part of the graphic object	Note 5 (See *Notes, p. 335*)
Major scale gradation	Number of major gradations (marked) in the scale	0 to 100
Minor scale gradation	Number of minor gradations (not marked) in the scale	0 to 100
Scale precision	Number of decimal places shown for the scale gradations (set to -1 to use a general exponential format)	-1 to 6
Maximum EU Value	Maximum value of the direct address for scaling, in physical units	
Minimum EU Value	Minimum value, in physical units, of the direct address for scaling	

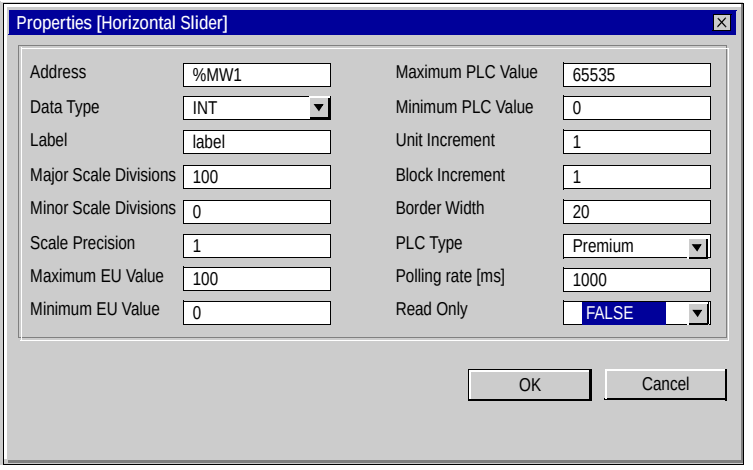
Property	Description	Limits
Maximum PLC Value	Gross maximum value (without scale) of the direct address in the PLC	Note 3 (See *Notes, p. 335*)
Minimum PLC Value	Gross minimum value (without scale) of the direct address in the PLC	Note 3 (See *Notes, p. 335*)
Limit Deadband[%]	Neutral range (as a percentage of the UP range) to apply to verification of the High/Low limit	0 to 10
High High Limit Value	Value expressed in physical units of the "High High" limit	
High Limit Value	Value expressed in physical units of the "High" limit	
Low Limit Value	Value expressed in physical units of the "Low" limit	
Low Low Limit Value	Value expressed in physical units of the "Low Low" limit	
High High Limit Value Color	Color of the indicator bar if the scale value is greater than the "High High" limit	
High Limit Value Color	Color of the indicator bar if the scale value is greater than the "High" limit	
Low Limit Value Color	Color of the indicator bar if the scale value is less than the "Low" limit	
Low Low Limit Value Color	Color of the indicator bar if the scale value is less than the "Low Low" limit	
Border Width	Width of border	
PLC Type	Type of PLC	Quantum or Premium
Polling rate [ms]	Scanning value	

The HTML code with the parameters of the widget above is as follows:

```
<APPLET
codebase="/classes"
archive="mbclient.jar,widgetslite.jar"
code="com.schneiderautomation.factorycast.gateway.
indicators.LiveHorizontalIndicatorApplet"
width="180" height="160">
   <PARAM name="progressbar"    value="true">
   <PARAM name="progresscolor"    value="#000000">
   <PARAM name="debug"    value="0">
   <PARAM name="BACKGRND"    value="LT_GRAY">
   <PARAM name="address"    value="1">
   <PARAM name="datatype"    value="REGISTER">
   <PARAM name="label"    value="label">
   <PARAM name="majorTics"    value="100">
   <PARAM name="minorTics"    value="0">
   <PARAM name="precision"    value="1">
   <PARAM name="maximum"    value="100">
   <PARAM name="minimum"    value="0">
   <PARAM name="maxValue"    value="65535">
   <PARAM name="minValue"    value="0">
   <PARAM name="borderWidth"    value="25">
   <PARAM name="limitHiHi"    value="99">
   <PARAM name="limitHi"    value="95">
   <PARAM name="limitLo"    value="5">
   <PARAM name="limitLoLo"    value="4">
   <PARAM name="deadband"    value="1">
   <PARAM name="rate"    value="1000">
   <PARAM name="unitId"    value="0">
   <PARAM name="colorHiHi"    value="RED">
   <PARAM name="colorHi"    value="ORANGE">
   <PARAM name="colorLoLo"    value="BLUE">
   <PARAM name="colorLo"    value="CYAN">
</APPLET>
```

Horizontal or Vertical Slider Setup

A horizontal or vertical slider gives an analog representation of the direct address of a Modbus device. It is a cursor that is proportional to the value, and it represents a percentage of its range in physical units. Using the mouse, you can change the value of the slider and trigger, and you can send a new value to the Modbus slave.



The properties of the slider are as follows:

Property	Description	Limits
Address	Address of the PLC variable	Note 1 (See *Notes, p. 335*)
Data type	Data type of the PLC address	Note 2 (See *Notes, p. 335*)
Label	Label displayed as part of the graphic object	Note 5 (See *Notes, p. 335*)
Major Scale Division	Number of major gradations (marked) in the scale	0 to 100
Minor Scale Division	Number of minor gradations (not marked) in the scale	0 to 100
Scale Precision	Number of decimal places shown for the scale gradations (set to -1 to use a general exponential format)	-1 to 6
Maximum EU value	Maximum value, in physical units, of the direct address for scaling	
Minimum EU Value	Minimum value of the direct address for scaling, in physical units	
Maximum PLC Value	Gross maximum value (without scale) of the direct address in the PLC	Note 3 (See *Notes, p. 335*)

Property	Description	Limits
Minimum PLC Value	Gross minimum value (without scale) of the direct address in the PLC	Note 3 (See *Notes, p. 335*)
Unit Increment	Amount by which the scale value is modified when you click on the slider arrows	
Block Increment	Amount by which the scale value is modified when you click on the slide area of the bar	
Border Width	Width (in pixels) of the border of the graphic object	
PLC Type	Type of PLC	Premium or Quantum
Polling rate [ms]	Scanning value	
Read Only	TRUE = read value, FALSE = read/write value	

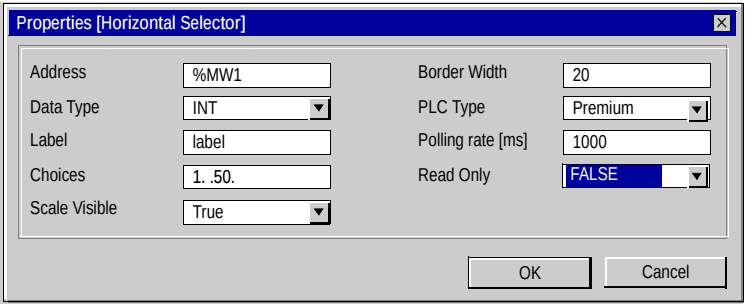
The HTML code with the parameters of the widget above is as follows:

```
<APPLET
codebase="/classes"
archive="mbclient.jar,widgetslite.jar"
code="com.schneiderautomation.factorycast.gateway.sliders.
LiveHorizontalSliderApplet"
width="180" height="160">
   <PARAM name="progressbar"    value="true">
   <PARAM name="progresscolor"    value="#000000">
   <PARAM name="debug"    value="0">
   <PARAM name="BACKGRND"    value="LT_GRAY">
   <PARAM name="address"    value="1">
   <PARAM name="datatype"    value="REGISTER">
   <PARAM name="label"    value="label">
   <PARAM name="majorTics"    value="100">
   <PARAM name="minorTics"    value="0">
   <PARAM name="precision"    value="1">
   <PARAM name="maximum"    value="100">
   <PARAM name="minimum"    value="0">
   <PARAM name="maxValue"    value="65535">
   <PARAM name="minValue"    value="0">
   <PARAM name="unitIncrement"    value="1">
   <PARAM name="blockIncrement"    value="1">
   <PARAM name="rate"    value="1000">
   <PARAM name="unitId"    value="0">
   <PARAM name="borderWidth"    value="20">
   <PARAM name="readOnly"    value="False">
</APPLET>
```

Horizontal or Vertical Selector

A horizontal or vertical selector allows the user to choose from a number of options. Once the selection has been made, the value corresponding to the choice is sent to the PLC. The choices are represented by marks on a "scale", the current selection being indicated by the position of the cursor on a slider.

Setup:



The properties of the slider are as follows:

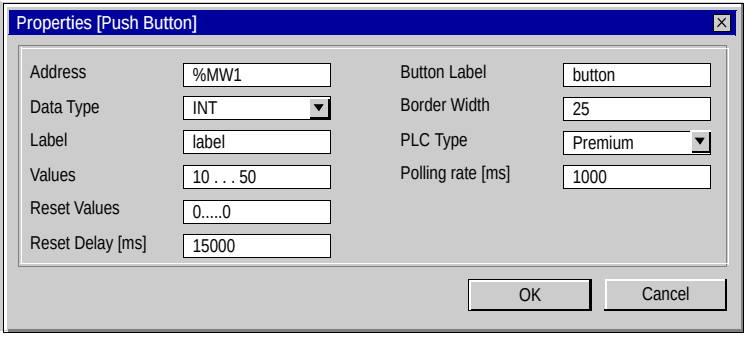
Property	Description	Limits
Address	Address of the PLC variable	Note 1 (See *Notes, p. 335*)
Data type	PLC address data type	Note 2 (See *Notes, p. 335*)
Label	Label displayed as part of the graphic object	Note 5 (See *Notes, p. 335*)
Choices	Setpoints to apply to the address value	
Border Width	Width (in pixels) of the border of the graphic object	
PLC Type	Type of PLC	Premium or Quantum
Polling rate [ms]	Scanning value	
Read Only	True = read value, False = read/write value	

The HTML code with the parameters of the widget above is as follows:

```
<APPLET
codebase="/classes"
archive="mbclient.jar,widgetslite.jar"
code="com.schneiderautomation.factorycast.gateway.selectors
.LiveHorizontalSelectorApplet"
width="180" height="160">
   <PARAM name="progressbar"    value="true">
   <PARAM name="progresscolor"    value="#000000">
   <PARAM name="debug"    value="0">
   <PARAM name="BACKGRND"    value="LT_GRAY">
   <PARAM name="address"    value="1">
   <PARAM name="datatype"    value="REGISTER">
   <PARAM name="label"    value="label">
   <PARAM name="scaleVisible"    value="True">
   <PARAM name="choices"    value="1=1000,9=9000,50=50000">
   <PARAM name="rate"    value="1000">
   <PARAM name="unitId"    value="0">
   <PARAM name="borderWidth"    value="20">
   <PARAM name="readOnly"    value="False">
</APPLET>
```

Push Button Setup

You can send a preset value to one or more Modbus slaves with a push button. A push button is activated with the mouse.



The properties of the push button are as follows:

Property	Description	Limits
Address	Address of the PLC variable	Note 1 (See *Notes, p. 335*)
Data type	PLC address data type	Note 2 (See *Notes, p. 335*)
Label	Label displayed as part of the graphic object	Note 5 (See *Notes, p. 335*)
Values	Setpoints to apply to the address value when the button is pressed	Note 4 (See *Notes, p. 335*)
Reset Values	Value to apply when the Reset Delay ends	
Reset Delay [ms]	Time in ms counted down after the button is pressed	
Button Label	Label to display on the button	
Border Width	Width (in pixels) of the border of the graphic object	
PLC Type	Type of PLC	Premium or Quantum
Polling rate [ms]	Scanning value	

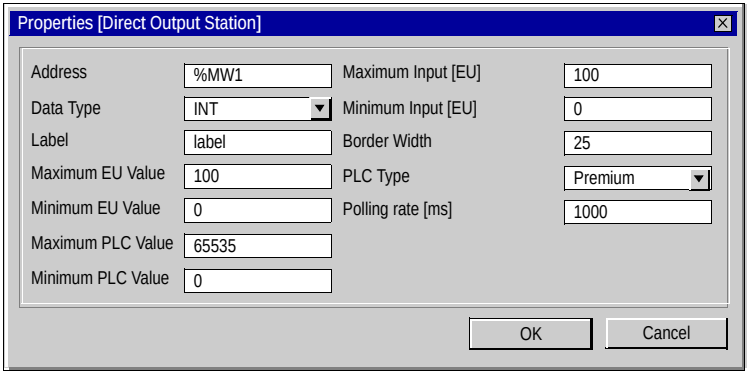
The HTML code with the parameters of the widget above is as follows:

```
<APPLET
codebase="/classes"
archive="mbclient.jar,widgetslite.jar"
code="com.schneiderautomation.factorycast.gateway.pushButton
.LivePushButtonApplet"
width="180" height="160">
   <PARAM name="progressbar"    value="true">
   <PARAM name="progresscolor"    value="#000000">
   <PARAM name="debug"    value="0">
   <PARAM name="BACKGRND"    value="LT_GRAY">
   <PARAM name="address"    value="1">
   <PARAM name="datatype"    value="REGISTER">
   <PARAM name="label"    value="label">
   <PARAM name="values"    value="10,50">
   <PARAM name="resetValues"    value="0">
   <PARAM name="resetDelay"    value="15000">
   <PARAM name="borderWidth"    value="25">
   <PARAM name="buttonLabel"    value="button">
   <PARAM name="rate"    value="1000">
   <PARAM name="unitID"    value="0">
</APPLET>
```

Note: In this example, when you press the button, the value 10 is applied to address 1 and the value 50 is applied to address 2 for 15,000 ms. After 15,000 ms, the two addresses are reset to 0.

Direct Output Window

With the Direct Output Window applet, you can enter a value in a text input field directly from the keyboard. If you enter a numerical value between the upper and lower preset limits, an OK button is activated. The value is sent to the Modbus slave each time you click OK or press the ENTER key (if the input field is active for keyboard input).



The direct output properties are as follows:

Property	Description	Limits
Address	Address of the PLC variable	Note 1 (See *Notes, p. 335*)
Data type	PLC address data type	Note 2 (See *Notes, p. 335*)
Label	Label displayed as part of the graphic object	Note 5 (See *Notes, p. 335*)
Maximum EU Value	Maximum value, in physical units, of the direct address for scaling	
Minimum EU Value	Minimum value, in physical units, of the direct address for scaling	
Maximum PLC Value	Gross maximum value (without scale) of the direct address in the PLC	Note 3 (See *Notes, p. 335*)
Minimum PLC Value	Gross minimum value (without scale) of the direct address in the PLC	Note 3 (See *Notes, p. 335*)
Maximum Input [EU]	Maximum setpoint for the input	
Minimum Input [EU]	Minimum setpoint for the input	
Border Width	Width (in pixels) of the border of the graphic object	
PLC Type	Type of PLC	Premium or Quantum

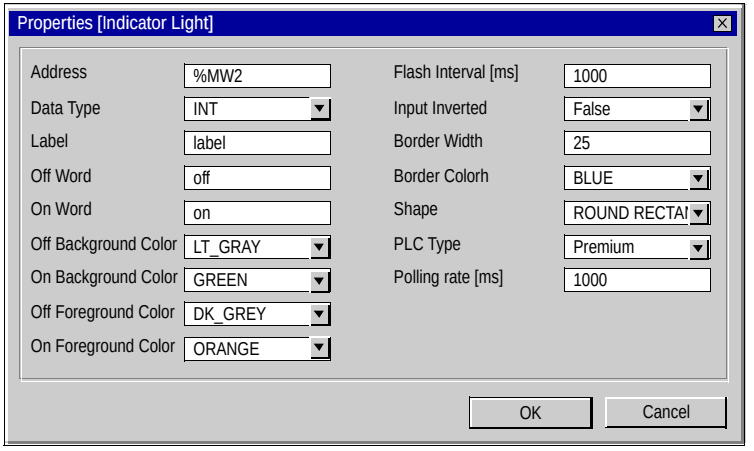
Property	Description	Limits
Polling rate [ms]	Scanning value	

The HTML code with the parameters of the widget above is as follows:

```
<APPLET
codebase="/classes"
archive="mbclient.jar,widgetslite.jar"
code="com.schneiderautomation.factorycast.gateway.direct
Output.LiveDirectOutputApplet"
width="180" height="160">
   <PARAM name="progressbar"    value="true">
   <PARAM name="progresscolor"    value="#000000">
   <PARAM name="debug"    value="0">
   <PARAM name="BACKGRND"    value="LT_GRAY">
   <PARAM name="address"    value="1">
   <PARAM name="datatype"    value="REGISTER">
   <PARAM name="label"    value="label">
   <PARAM name="maximum"    value="100">
   <PARAM name="minimum"    value="0">
   <PARAM name="maxValue"    value="65535">
   <PARAM name="minValue"    value="0">
   <PARAM name="maxInputValue"    value="100">
   <PARAM name="minInputValue"    value="0">
   <PARAM name="rate"    value="1000">
   <PARAM name="unitId"    value="0">
   <PARAM name="borderWidth"    value="25">
</APPLET>
```

Indicator Light Setup

The indicator light provides a dual indication of the value of a direct address in a PLC. If the Input inverted property is not set to TRUE, a 0 input value is declared OFF, and a non-zero value is declared ON. If the Flash Interval property is set to a positive value, the indicator light flashes when the input value is equivalent to ON.



The properties of the indicator light are as follows:

Property	Description	Limits
Address	Address of the PLC variable	Note 1 (See *Notes, p. 335*)
Data type	PLC address data type	Note 2 (See *Notes, p. 335*)
Label	Label displayed as part of the graphic object	Note 5 (See *Notes, p. 335*)
OFF Word	Text displayed when the input value is OFF	
ON Word	Text displayed when the input value is ON	
OFF Background Color	Background color of the indicator light when OFF Word is displayed	
ON Background Color	Background color of the indicator light when ON Word is displayed	
OFF Foreground Color	OFF Word text color	
ON Foreground Color	ON Word text color	
Flash Interval	The flashing time for the indicator light (expressed in ms) when the input value is ON. Set to zero for no flashing	200 to 2000

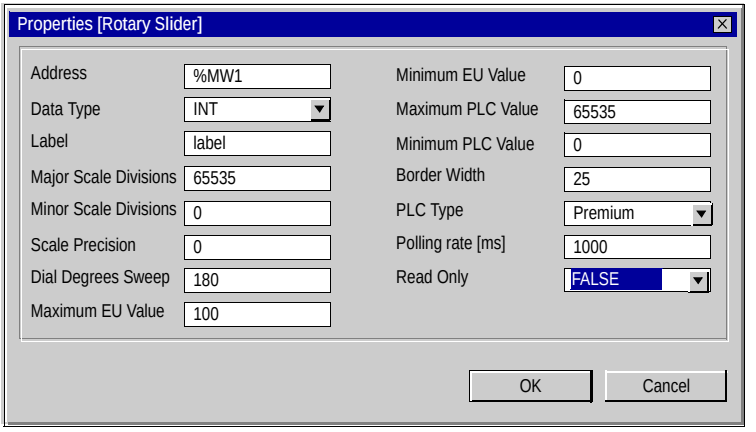
Property	Description	Limits
Input inverted	On TRUE, inverts the input value (the indicator light displays the OFF Word when the input value is ON)	
Border Width	Width (in pixels) of the border of the graphic object	
Border Color	Color of the border	
Shape	Shape (circular, rectangular, etc.) of the indicator light	
PLC Type	Type of PLC	Premium or Quantum
Polling rate [ms]	Scanning value	

The HTML code with the parameters of the widget above is as follows:

```
<p><APPLET
codebase="/classes"
archive="mbclient.jar,widgetslite.jar"
code="com.schneiderautomation.factorycast.gateway.indica-
torLight.LiveIndicatorLightApplet"
width="180" height="160">
   <PARAM name="progressbar"    value="true">
   <PARAM name="progresscolor"    value="#000000">
   <PARAM name="debug"    value="0">
   <PARAM name="BACKGRND"    value="LT_GRAY">
   <PARAM name="address"    value="2">
   <PARAM name="datatype"    value="REGISTER">
   <PARAM name="label"    value="label">
   <PARAM name="offWord"    value="off">
   <PARAM name="onWord"    value="on">
   <PARAM name="offWordBackground"    value="LT_GRAY">
   <PARAM name="onWordBackground"    value="GREEN">
   <PARAM name="offWordForeground"    value="DK_GRAY">
   <PARAM name="onWordForeground"    value="ORANGE">
   <PARAM name="flashInterval"    value="1000">
   <PARAM name="inputInverted"    value="False">
   <PARAM name="borderWidth"    value="25">
   <PARAM name="borderColor"    value="BLUE">
   <PARAM name="shape"    value="ROUND RECTANGLE">
   <PARAM name="rate"    value="1000">
   <PARAM name="unitId"    value="0">
</APPLET>
```

Rotary Slider Setup

A rotary slider gives an analog representation of the direct address of a Modbus device. It draws a position proportional to the value of the address and represents a percentage of its range in physical units on a circular dial. The size of the circular dial (cycle in degrees of a circle) and the button color can be configured. Using the mouse, the user can change the value of the rotary slider and trigger sending a new value to the PLC.



The properties of the slider are as follows:

Property	Description	Limits
Address	Address of the PLC variable	Note 1 (See *Notes, p. 335*)
Data type	PLC address data type	Note 2 (See *Notes, p. 335*)
Label	Label displayed as part of the graphic object	Note 5 (See *Notes, p. 335*)
Major scale gradation	Number of major gradations (marked) in the scale	0 to 100
Minor scale gradation	Number of minor gradations (not marked) in the scale	0 to 100
Scale precision	Number of decimal places in the scale gradations (set to -1 to use a general exponential format)	-1 to 6
Dial Degrees Sweep	Portion of circular dial used to draw the scale	60 to 300
Maximum EU Value	Maximum value of the direct address for scaling, in physical units	
Minimum EU Value	Minimum value of the direct address for scaling, in physical units	

Property	Description	Limits
Maximum PLC Value	Gross maximum value (without scale) of the direct address in the PLC	Note 3 (See *Notes, p. 335*)
Minimum PLC Value	Gross minimum value (without scale) of the direct address in the PLC	Note 3 (See *Notes, p. 335*)
Border Width	Width (in pixels) of the border of the graphic object	
PLC Type	Type of PLC	Premium or Quantum
Polling rate [ms]	Scanning value	
Read Only	True = read value, False = read/write value	

The HTML code with the parameters of the widget above is as follows:

```
<APPLET
codebase="/classes"
archive="mbclient.jar,widgetslite.jar"
code="com.schneiderautomation.factorycast.gateway.sliders.
LiveRotosliderApplet"
width="180" height="160">
   <PARAM name="progressbar"    value="true">
   <PARAM name="progresscolor"    value="#000000">
   <PARAM name="debug"    value="0">
   <PARAM name="BACKGRND"    value="LT_GRAY">
   <PARAM name="address"    value="1">
   <PARAM name="datatype"    value="REGISTER">
   <PARAM name="label"    value="label">
   <PARAM name="majorTics"    value="65535">
   <PARAM name="minorTics"    value="0">
   <PARAM name="precision"    value="0">
   <PARAM name="degSweep"    value="180">
   <PARAM name="maximum"    value="100">
   <PARAM name="minimum"    value="0">
   <PARAM name="maxValue"    value="65535">
   <PARAM name="minValue"    value="0">
   <PARAM name="borderWidth"    value="25">
   <PARAM name="rate"    value="1000">
   <PARAM name="readOnly"    value="False">
   <PARAM name="unitID"    value="0">
```

Notes

1.	The address can be:	
	%MW	IEC internal word
	%MD	IEC double word
	%M	IEC internal bit
	400000	Concept integer
	100000	Concept boolean
2.	The various values of the Data type property have the following meaning:	
	Data type	Meaning
	`INT`	`integer`
	`DINT`	`double integer`
	`BOOL`	`boolean`
3.	The limits of the Maximum PLC Value and Minimum PLC Value properties are the natural limits of the configured Data type property.	
4.	Specify at least one value for a push button. If several values are entered, they will be assigned to an address table starting with the direct address indicated.	
5.	If you specify `param name="label" value="$data$'` in the HTML code, the applet displays the numerical value of the data in place of a label.	

Alarm Viewer

10

Alarm Viewer

Overview

The Alarm Viewer is a Java applet designed to monitor Premium and Quantum PLCs. It is not currently supported by Micro. It is a Client Runtime Function and will run in the JVM of internet client browser (Internet Explorer, Netscape Navigator, etc.).

References

Refer to the following manuals for a full explanation of the software.

Language	Manual	Part Number	Order Number
Spanish	Manual de instalación de las funciones de diagnióstico	TLX DS DIAG PL7 33S	W915905840701A02
German	Handbuch zur Inbetriebnahme der Diagnosefunktionen	TLX DS DIAG PL7 33G	W915905840201A02
French	Manuel de mise en œuvre des fonctions de diagnostic	TLX DS DIAG PL7 33F	W915905840101A02
English	Diagnostic Functions Setup Manual	TLX DS DIAG PL7 33E	W915905840301A02

Abbreviations

DFB: Derived Function Block

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Display	338
Operation and Management of Alarms	341
Limitations	342

Display

Overview

This section discusses setting and interpreting the display of the Alarm Viewer.

Access

Select **Diagnostics | Alarm Viewer**.

Note: Alarm Viewer is a plug-in. You must download it before you can use it.

The display is composed of:
- button bar (in the work area),
- list of alarms,
- status frame.

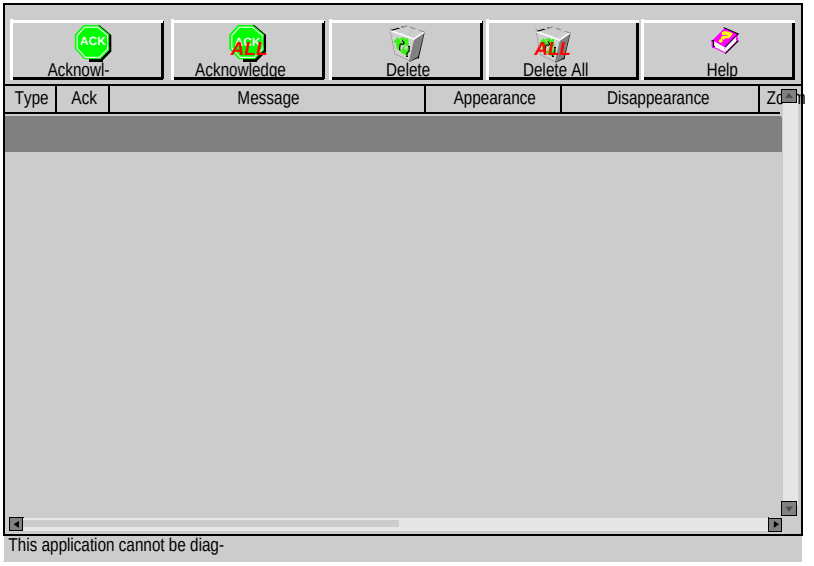
Button Bar

The button bar contains five buttons.

	This button is used to acknowledge a selected alarm in the list. A request is sent to the PLC (Diagnostic Buffer).
	This button is used to acknowledge all alarms in the list that can be acknowledged. A request for each alarm is sent to the PLC.
	This button is used to delete a selected alarm in the list. There is no request sent to the PLC. This command affects only the alarm list in Alarm Viewer.
	This button is used to delete all alarms in the list that can be deleted. There is no request sent to the PLC. This command affects only the alarm list in Alarm Viewer.
	This button displays a frame that contains help.

List of Alarms

Alarms in the list are displayed in historical order. The last alarm in runtime is added to the end of the list.



Each line displayed in the list corresponds to an alarm and contains the following information

Type: An icon that represents the alarm type. For each type of alarm there is a different icon.



DFB alarm



Grafcet alarm



System alarm



A-SI alarm

Ack: An icon that represents the acknowledgement status.

Message: Alarm text.

Appearance: Date and time when alarm occurs.

Disappearance: Date and time when alarm disappears.

Zone or area number: Area or geographical zone from which the error comes (common area: 0).

Note: Consider the following.

- You can change the column width by using the mouse.
- The number of alarms that can be displayed in the list is limited to 1000. When this limit is reached, a warning message is displayed in the status frame. Alarm Viewer recognizes subsequent alarms, but doesn't display them. To see the next alarms, the user must purge the list of alarms displayed by Alarm Viewer.
- Alarm Viewer displays alarms from all zones. The zone contains values from 0 ... 15.

Note: An alarm that appears is displayed in red text and there is no **Disappearance**. An alarm that disappears is displayed in green text with **Disappearance**.

Status Frame

This frame is used to display the error, an information message, or throughput messages. For example a message such as, "Diagnostic Application is not configured in this application."

Operation and Management of Alarms

Browsing

Use the UP, DOWN, PAGEUP, or PAGEDOWN keys or the mouse to select the alarms in the list. Use the scroll bar if the list contains more alarms than can be displayed in the dialog.

Acknowledgment

To acknowledge an alarm that requires acknowledgment, select the alarm and use the appropriate toolbar button.

Several alarms can be acknowledged at one time by using the **Ack All** button.

Note: An alarm can be acknowledged by another Alarm Viewer. In this case, Alarm Viewer is notified and the alarm is displayed as acknowledged.

Deleting an Alarm from the List

- An alarm that requires acknowledgment or that has not disappeared cannot be deleted.
- DELETE and DELETE ALL buttons can be used to delete only those alarms that have disappeared and have been acknowledged (if acknowledgment is required).

Limitations

Overview

For each alarm, there is additional information stocked in the diagnostic buffer. For example, many DFBs have outputs named STATUS (word) where the error cause is coded. Alarm Viewer doesn't use this information. It displays only basic information about alarms.

Alarm Viewer Functionality

The Alarm Viewer works only under these conditions.

Product	Firmware Platform	Software Platform
TSX ETY 110 WS	Premium PLC TSX57/PCX57/PMX57 V3.3	PL7 PRO PL-7 Junior V3.3 Unity Pro
TSX ETY 5103	Premium PLC TSX57/PCX57/PMX57 V3.3	PL7 PRO PL-7 Junior V3.3 Unity Pro
140 NOE 771 11	Quantum Unity Platform	Unity Pro

Appendices



Appendices for FactoryCast User's Guide

Overview

There are two sets of appendices for the FactoryCast User's Guide. The first appendix is browser requirements, settings, and security considerations, and the second appendix is performance benchmarks.

What's in this Appendix?

The appendix contains the following chapters:

Chapter	Chapter Name	Page
A	Browser Requirements, Settings, and Security Considerations	345
B	Performance Benchmarks	349
C	SOAP Web Services	369

Browser Requirements, Settings, and Security Considerations

A

Browser Requirements, Settings, and Security Considerations

Overview

In order to view the Java applets on FactoryCast Web sites, you must have the correct browser version. This appendix discusses that requirement and other browser considerations.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Browser Version	346
Browser Settings	347
Browser Security Considerations	348

Browser Version

Overview

To display Java applets in the FactoryCast Web pages, you must have a browser that supports Java 2 version 1.4.x or higher. It is recommended that you use the Sun Java plug-in, available at the Web site: http://java.sun.com. Choose download and install the J2SE platform.

Which Browsers Qualify?

These browsers meet the requirements:

- Netscape Communicator 4.06 and higher,
- Internet Explorer 4.0 w/Service Pack 2 and higher.

Which Browser Do I Have?

Your browser name and version are displayed at the bottom of the FactoryCast Home Page. If you need to upgrade your browser, the Internet addresses for the two major browsers are listed below. (Upgrades are free.You only need to download the latest version.)

How Do I Upgrade?

Supported browsers are listed below:

Download the browser:	**At Web Site:**
Netscape Communicator (Includes the browser Netscape Navigator)	http://www.netscape.com
Internet Explorer	http://www.microsoft.com

Browser Settings

Overview

If you are using Microsoft Internet Explorer and you have trouble viewing Java applets, you can modify your browser security settings to improve your ability to view the applets.

Modifying Security Settings

Follow the steps in the table below to modify the security settings of the browser.

Step	Action
1	Open Internet Explorer.
2	Select **Tools | Internet Options** from the menu bar. **Result:** The **Internet Options** window appears.
3	Select the **Security** tab.
4	Select the **Custom Level** button in the **Security Level** settings section.
5	Select the **Settings** button. **Result:** The Security Settings dialog appears.
6	Scroll down until **Microsoft VM** is found under **Java Permissions**. Click the **Custom** radio button. Result: The **Java Custom Settings** button appears.
7	Select the **Java Custom Settings** button. **Result:** The Internet Zone dialog appears.
8	Select the **Edit Permissions** tab.
9	Under the **Run Unsigned | Content**, select **Enable**.
10	Click **OK**.

Browser Security Considerations

Overview	Both Netscape Navigator and Internet Explorer remember a user name and password once entered for a Web site.
We Recommend	Close the browser after each session to prevent unauthorized people from using your passwords to gain read or write access to your site. If you are using Internet Explorer, you may also have to change your security settings using the custom settings option. Under the options for User Authentication, select "Prompt for user name and password."

Performance Benchmarks

B

Performance Benchmarks

Overview

This appendix contains performance benchmarks for several configurations.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
B.1	Micro Performance Benchmarks	350
B.2	Premium Performance Benchmarks	355
B.3	Quantum Performance Benchmarks	362
B.4	FactoryCast Configurator Performance Benchmarks	367

B.1 Micro Performance Benchmarks

Micro Performance

Overview

This section describes the performance of the TSX ETZ 510 embedded Web server.

What's in this Section?

This section contains the following topics:

Topic	Page
Micro Performance Benchmarks	351
Monitoring Applet Performance	352
SOAP Performance Benchmarks	354

Micro Performance Benchmarks

Overview

This section contains information on the impact that access to the embedded Web server has on the TSX ETY 110 messaging module's performances.

All response times are in ms. The client and the server have the same cycle time.

Average time of 1 EF Read of 50 words

	Cyclic	**period 20ms**	**period 50ms**	**period 100ms**
TSX ETZ 510 **Client** 1 EF running	140	150	150	190
TSX ETZ 510 **Client** 1 EF running with Data Editor	190	200	220	290
TSX ETZ 510 **Client** 1 EF running with Sysdiag (Rack Viewer)	150	150	190	270
TSX ETZ 510 **Client and Server** 1 EF running with Sysdiag (Rack Viewer)	190	190	220	300

Average time of 3 EF Read of 50 words

	Cyclic	**period 20ms**	**period 50ms**	**period 100ms**
TSX ETZ 510 **Client** 1 EF running	300	300	300	300
TSX ETZ 510 **Client** 1 EF running with Data Editor	400	400	420	500
TSX ETZ 510 **Client** 1 EF running with Sysdiag (Rack Viewer)	340	350	360	530
TSX ETZ 510 **Client and Server** 1 EF running with Sysdiag (Rack Viewer)	530	500	510	610

Monitoring Applet Performance

Ethernet Configuration

- PC Intel Pentium IV 1.7 GHz,
- JVM 1.4.2_09 (Java virtual machine),
- 100 Mb/s Ethernet.

Modem Configuration

- PC Intel Pentium IV 1.7 GHz,
- JVM 1.4.2_09,
- 56k modem (38400 bauds).

Data Editor

This table displays Data Editor performances:

RDE	100 Mb/s	Modem
1st page loading	2 s	2 m 55 s
Following loading	2 s	14 s
Time refresh value, full table	300 ms	300 ms

Data Editor Lite

This table displays Data Editor Lite performances:

RDE	Modem
1st page loading	1 m 20 s
Following loading	4 s
Time refresh value, full table	300 ms

Graphic Editor

This table displays Graphic Editor performances:

GDE	100 Mb/s	Modem
First page loading without graphic page	3 s	5 m 46 s
Following loading	2 s	20 s
Loading page with 20 widgets	2 s	11 s
Loading page with 50 widgets	2 s	18 s
Loading page with 100 widgets	2 s	37 s
Loading page with 200 widgets	3 s	1 m 05 s

Custom Pages

Concerning Custom Pages performances with widgets refer to the previous table.

This table displays Custom Pages performances with new widgets lite:

Custom pages	Modem
First page loading with 20 widgets lite	55 s
Following loading page with 20 widgets lite	3 s
First page loading with 50 widgets lite	1 m 04 s
Following loading page with 50 widgets lite	4 s
First page loading with 100 widgets lite	1 m 30 s
Following loading page with 100 widgets lite	5 s
First page loading with 200 widgets lite	1 m 45 s
Following loading page with 200 widgets lite	8 s

SOAP Performance Benchmarks

SOAP Communication

Microsoft Application Center test tools (supplied with Visual Studio.net 2003) were used for the performance measurements.

The following table shows measured performance:

Nature of measurement	Time spent for 100 iterations	Average request rate
The request reads 50 registers (from %MW0 to %MW49)	28 s	3.58/s
The request writes 50 registers (one Modbus request)	27 s	3.7/s
The request reads 50 symbols of noncontiguous registers	7 m 14 s	0.23/s
The request reads 50 symbols of contiguous registers	35 s	2.9/s

B.2 Premium Performance Benchmarks

Premium Performance

Overview

This section describes the performance of the Premium embedded Web server.

What's in this Section?

This section contains the following topics:

Topic	Page
Premium Performance Benchmarks	356
Monitoring Applet Performance on ETY5103	359
SOAP Performance Benchmarks	361

Premium Performance Benchmarks

Overview

This section contains information on the impact that access to the embedded Web server has on the TSX ETY 110 messaging module's performances.

All the response times are in ms. The client and the server have the same cycle time.

Average time of 1 EF READ-VAR of 50 words

ETHWAY

	Cyclic	**5**	**10**	**20**	**50**
TSX ETY 110	73	73	76	82	103
TSX ETY 110 WS installed	73	72	76	83	138
TSX ETY 110 WS running with data editor	73	73	75	83	127
TSX ETY 110 WS running with sysdiag on ETY client	86	88	91	98	149
TSX ETY 110 WS running with sysdiag on ETY client and server	93	98	99	99	148

TCP/IP

	Cyclic	**5**	**10**	**20**	**50**
TSX ETY 110	74	75	77	84	102
TSX ETY 110 WS installed	74	73	77	84	106
TSX ETY 110 WS running with data editor	74	73	77	85	108
TSX ETY 110 WS running with sysdiag on ETY client	80	85	90	98	109
TSX ETY 110 WS running with sysdiag on ETY client and server	92	94	100	108	128

Average time of 8 EF READ-VAR of 50 words

ETHWAY

	Cyclic	**10**	**50**
TSX ETY 110	221	229	247
TSX ETY 110 WS installed	223	229	247
TSX ETY 110 WS running with data editor	224	231	249
TSX ETY 110 WS running with sysdiag on ETY client	236	240	254
TSX ETY 110 WS running with sysdiag on ETY client and server	246	252	275

TCP/IP

	Cyclic	**10**	**50**
TSX ETY 110	244	254	287
TSX ETY 110 WS installed	244	261	291
TSX ETY 110 WS running with data editor	245	259	293
TSX ETY 110 WS running with sysdiag on ETY client	262	270	309
TSX ETY 110 WS running with sysdiag on ETY client and server	304	307	337

Throughput Time of Bridge

ETHWAY

	Cyclic	10	20	50
TSX ETY 110	55	56	55	26
TSX ETY 110 WS running	51	51	47	18
TSX ETY 110 WS running with Comm ETY Bridge	61	58	58	56
TSX ETY 110 WS running with Comm ETY client and server	62	63	62	23

TCP/IP

	Cyclic	10	20	50
TSX ETY 110	57	57	60	31
TSX ETY 110 WS running	58	56	55	33
TSX ETY 110 WS running with Comm ETY Bridge	65	63	64	67
TSX ETY 110 WS running with Comm ETY client and server bridge	66	69	67	54

Monitoring Applet Performance on ETY5103

Ethernet Configuration

- PC Intel Pentium IV 1.7 GHz,
- JVM 1.4.2_09 (Java virtual machine),
- 100 Mb/s Ethernet.

Modem Configuration

- PC Intel Pentium IV 1.7 GHz,
- JVM 1.4.2_09,
- 56k modem (38400 bauds).

Data Editor

This table displays Data Editor performances:

RDE	100 Mb/s	Modem
1st page loading	2 s	2 m 55 s
Following loading	2 s	14 s
Time refresh value, full table	300 ms	300 ms

Data Editor Lite

This table displays Data Editor Lite performances:

RDE	Modem
1st page loading	1 m 20 s
Following loading	4 s
Time refresh value, full table	300 ms

Graphic Editor

This table displays Graphic Editor performances:

GDE	100 Mb/s	Modem
First page loading without graphic page	3 s	5 m 46 s
Following loading	2 s	20 s
Loading page with 20 widgets	2 s	11 s
Loading page with 50 widgets	2 s	18 s
Loading page with 100 widgets	2 s	37 s
Loading page with 200 widgets	3 s	1 m 05 s

Custom Pages

Concerning Custom Pages performances with widgets refer to the previous table.

This table displays Custom Pages performances with new widgets lite:

Custom pages	Modem
First page loading with 20 widgets lite	55 s
Following loading page with 20 widgets lite	3 s
First page loading with 50 widgets lite	1 m 04 s
Following loading page with 50 widgets lite	4 s
First page loading with 100 widgets lite	1 m 30 s
Following loading page with 100 widgets lite	5 s
First page loading with 200 widgets lite	1 m 45 s
Following loading page with 200 widgets lite	8 s

SOAP Performance Benchmarks

SOAP communication

Microsoft Application Center test tools (supplied with Visual Studio.net 2003) were used for the performance measurements.

The following table shows measured performance:

Nature of measurement	Time spent for 100 iterations	Average request rate
The request reads 50 registers (from %MW0 to %MW49)	20 s	5/s
The request writes 50 registers (one Modbus request)	20 s	5/s
The request reads 50 symbols of noncontiguous registers	4 m 09 s	0.30/s
The request reads 50 symbols of contiguous registers	27 s	3.7/s

B.3 Quantum Performance Benchmarks

Quantum Performance

Overview

This section describes the performance of the Quantum embedded Web server.

What's in this Section?

This section contains the following topics:

Topic	Page
Quantum Performance Benchmarks	363
Performance on NOE77111	364
SOAP Performance Benchmarks	366

Quantum Performance Benchmarks

Overview

These Quantum Web Embedded Server performance tests were run on Win95 on a Dell OptiPlex Gxi at 200 MHZ. The PLC was a Quantum 424.

Performance Data

The following table contains the performance data for the Quantum Web Embedded Server.

Web Scanner w/MSTR Block	50 scans	100 scans	200 scans	500 scans
Web scanner loaded, 1 MSTR	48.2ms	33.4ms 134.3ms	34.3ms	34.3ms
Web scanner loaded, 2 MSTR	43.7ms 66.4ms	45.6ms	43.8ms	46.7ms
Web scanner loaded, 4 MSTR	72.35ms 86.19ms	73ms	74ms	74ms 90ms
Web scanner loaded, 4 MSTR, 1 browser	107.3ms	72.5ms	73ms	72ms
Web scanner loaded, 4 MSTR, 3 browsers	78.9ms 94.2ms	80.5ms	81.1ms	120ms
Web scanner loaded, 4 MSTR, 5 browsers	105ms	80.5ms	80ms	90ms
Web scanner loaded, 4 MSTR, 5 instances of Internet Explorer (no cache)	104.1ms 92.5ms	85.2ms	88.9ms	90ms
Web scanner loaded, 4 MSTR, 5 instances of Internet Explorer (cache on)	108ms 83.7ms	85.2ms	90ms	120ms 92.5ms

Performance on NOE77111

Ethernet Configuration

- PC Intel Pentium IV 1.7 GHz,
- JVM 1.4.2_09 (Java virtual machine),
- 100 Mb/s Ethernet.

Modem Configuration

- PC Intel Pentium IV 1.7 GHz,
- JVM 1.4.2_09,
- 56k modem (38400 bauds).

Data Editor

This table displays Data Editor performances:

RDE	100 Mb/s	Modem
1st page loading	2 s	2 m 55 s
Following loading	2 s	14 s
Time refresh value, full table	300 ms	300 ms

Data Editor Lite

This table displays Data Editor Lite performances:

RDE	Modem
1st page loading	1 m 20 s
Following loading	4 s
Time refresh value, full table	300 ms

Graphic Editor

This table displays Graphic Editor performances:

GDE	100 Mb/s	Modem
First page loading without graphic page	3 s	5 m 46 s
Following loading	2 s	20 s
Loading page with 20 widgets	2 s	11 s
Loading page with 50 widgets	2 s	18 s
Loading page with 100 widgets	2 s	37 s
Loading page with 200 widgets	3 s	1 m 05 s

Custom Pages

Concerning Custom Pages performances with widgets refer to the previous table.

This table displays Custom Pages performances with new widgets lite:

Custom pages	Modem
First page loading with 20 widgets lite	55 s
Following loading page with 20 widgets lite	3 s
First page loading with 50 widgets lite	1 m 04 s
Following loading page with 50 widgets lite	4 s
First page loading with 100 widgets lite	1 m 30 s
Following loading page with 100 widgets lite	5 s
First page loading with 200 widgets lite	1 m 45 s
Following loading page with 200 widgets lite	8 s

SOAP Performance Benchmarks

SOAP Communication

Microsoft Application Center test tools (supplied with Visual Studio.net 2003) were used for the performance measurements.

The following table shows measured performance:

Nature of measurement	Time spent for 100 iterations	Average request rate
The request reads 50 registers (from %MW0 to %MW49)	21 s	4.77/s
The request writes 50 registers (one Modbus request)	22 s	4.85/s
The request reads 50 symbols of noncontiguous registers	4 m 59 s	0.33/s
The request reads 50 symbols of contiguous registers	31 s	3.22/s

B.4 FactoryCast Configurator Performance Benchmarks

FactoryCast Configurator Performance Benchmarks

Overview

This section provides test results for downloading a namespace and for downloading a Web site.

Application Download

In this performance test, the **Download Namespace Only** option was chosen.

Trial	ETY Embedded Server 110 WS (min:sec)	NOE Embedded Server 211 10 (min:sec)	NOE Embedded Server 771 10 (min:sec)
1	0:20	0:05	0:03
2	0:20	0:18	0:04
3	0:20	0:11	0:03

Web Site Download

In this test, 10 HTML files of 2 K each were downloaded to the server.

Trial	ETY Embedded Server 110 WS (min:sec)	NOE Embedded Server 211 10 (min:sec)	NOE Embedded Server 771 10 (min:sec)
1	1:32	1:25	0:06
2	2:00	2:45	0:06
3	2:00	2:45	0:06

SOAP Web Services

C

Presentaton

Purpose of this chapter

This chapter describes the various SOAP Web services implemented in FactoryCast Web servers. SOAP Web services are fully compliant with the W3C WS-I Web services standards. They provide a new efficient and standard means of making controller devices interact directly with IT/management level applications.

Web services are based on standards such as:

- SOAP (Single Object Access Protocol), the exchange protocol carried out via the HTTP (HyperText Transfer Protocol) channel.
- WSDL (Web Services Description Language), in XML format.
- XML (eXtensible Markup Language), the universal data exchange standard.

FactoryCast SOAP Web services act as SOAP server interfaces. They allow developers to easily design client applications that can exchange data directly with FactoryCast Web servers. Applications such as Microsoft.NET, SQL Server, Microsoft Office, IBM (WebSphere), SUN (Java, Eclipse), Lotus, Oracle, SAP, MES, ERP and so forth can be interfaced directly with FactoryCast using SOAP Web services.

Two types of web services are provided in FactoryCast modules as SOAP server interfaces:

- ModbusXMLDA: Web service to implement data access to Modbus variables
- SymbolicXMLDA: Web service to implement Symbolic data access

The Web services provided by FactoryCast are compatible with the WS-I basic profile 1.1.

Note: For more information on Web services, please refer to specialized publications on the subject.

Warning

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Principles for Designing a SOAP Client Interface	371
ModBusXMLDA SOAP Services	372
SymbolicXMLDA SOAP Services	383
How to optimize massive symbolic requests	387
Visual Basic examples	390
Online SOAP Documentation	391

Principles for Designing a SOAP Client Interface

Overview

An interface enables a SOAP client application to communicate directly with a FactoryCast Web server module. Exchanges are initiated by the SOAP client application. The FactoryCast server responds to these requests.

A SOAP Client Interface

The following table describes the process of designing a SOAP client interface:

Stage	Action
1	Create the client application: The development environment (for example, Visual Studio.net) connects to a FactoryCast Web server module where it can access a list of available Web services. The Web server returns descriptions of the requested services as WSDL objects.
2	Develop the client application: The developer integrates the Web service APIs using the code retrieved in the previous stage as a Web reference and generates the client application.
3	Execute the client application: In run mode, the client application communicates in real time with the FactoryCast Web server module using the SOAP protocol.

ModBusXMLDA SOAP Services

Overview

The ModbusXMLDA Web Service is implemented to provide data access to Modbus variables using a syntax similar to the Modbus protocol.

This Web service offers the following APIs:

- ReadDeviceIdentification,
- ReadMultipleRegisters,
- WriteMultipleRegisters,
- ReadCoils,
- WriteMultipleCoils,
- ReadInt32,
- WriteInt32.

Data amounts for API parameters

This table presents the maximum amount of data, according to the modules, that can be used for API parameters:

APIs	ETY 5103	ETZ 510	NOE 77111
ReadMultipleRegisters	125 (Register)	60 (Register)	124 (Register)
WriteMultipleRegisters	120 (Register)	56 (Register)	96 (Register)
ReadCoils	512 (Coils)	256 (Coils)	255 (Coils)
WriteMultipleCoils	511 (Coils)	255 (Coils)	254 (Coils)
ReadInt32	62 (Int32)	62 (Int32)	61 (Int32)
WriteInt32	59 (Int32)	58 (Int32)	48 (Int32)

ReadDevice-Identification

This request returns the entire device identification of the destination specified by the ID provided in the request.

This API is mapped on Modbus Function Code 43 14.

Parameters

Input: int UnitID

Output: string VendorName, ProductCode, MajorMinorRevision, VendorURL, ProductName, ModelName, UserApplicationName, TRImplementationClass, TRCommunicationServices

Example The following is an example of a SOAP request and its response:

```
POST /ws/ModbusXmlDa
HTTP/1.0 Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-electric.com/ws/
ModbusXmlDa/ReadDeviceIdentification"
<?xml version="1.0" encoding="utf-8" ?> <soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
   <soap:Body>
     <ReadDeviceIdentification xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
       <UnitID>int</UnitID>
     </ReadDeviceIdentification>
   </soap:Body>
 </soap:Envelope>
HTTP/1.0 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length <?xml version="1.0" encoding="utf-8"
?><soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/
envelope/">
   <soap:Body>
     <ReadDeviceIdentificationResponse xmlns="http://
www.schneider-electric.com/ws/ModbusXmlDa/">
       <ReadDeviceIdentificationResult>
         <DeviceIdentification>
                 <VendorName>string</VendorName>
                 <ProductCode>string</ProductCode>
           <MajorMinorRevision>string</MajorMinorRevision>
           <VendorUrl>string</VendorUrl>
           <ProductName>string</ProductName>
           <ModelName>string</ModelName>
           <UserApplicationName>string</UserApplicationName>
           <TRImplementationClass>string</TRImplementa-
tionClass>
           <TRCommunicationServices>string</TRCommunication-
Services>
         </DeviceIdentification>
       </ReadDeviceIdentificationResult>
     </ReadDeviceIdentificationResponse>
   </soap:Body>
</soap:Envelope>
```

ReadMultiple-Registers	This is the API used to implement a read multiple registers action. The request provides three parameters: ● the destination of the read request, ● the starting address, ● the number of values to read (base index is provided by starting address). In response, the list of values is provided. This API is mapped on Modbus Function Code 03.
Parameters	Input: int UnitID, Address, Quantity Output: array of int Result

Example The following is an example of a SOAP request and its response:

```
POST /ws/ModbusXmlDa
 HTTP/1.0 Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-electric.com/ws/
ModbusXmlDa/ReadMultipleRegisters"
<?xml version="1.0" encoding="utf-8" ?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/
envelope/">
   <soap:Body>
     <ReadMultipleRegisters xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
       <UnitID>int</UnitID>
       <Address>int</Address>
       <Quantity>int</Quantity>
     </ReadMultipleRegisters>
   </soap:Body>
 </soap:Envelope>

HTTP/1.0 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length <?xml version="1.0" encoding="utf-8" ?>
 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/
envelope/">
   <soap:Body>
     <ReadMultipleRegistersResponse xmlns="http://
www.schneider-electric.com/ws/ModbusXmlDa/">
       <ReadMultipleRegistersResult>
         <int>int</int>
         <int>int</int>
       </ReadMultipleRegistersResult>
     </ReadMultipleRegistersResponse>
   </soap:Body>
 </soap:Envelope>
```

WriteMultiple-Registers	This is the API used to implement a write multiple registers action. The request provides three parameters: ● the destination of the write request, ● the starting address, ● the values to write. This API is mapped on Modbus Function Code 16.
Parameters	Input: int UnitID, Address, array of int Value Output: none

Example

The following is an example of a SOAP request and its response:

```
POST /ws/ModbusXmlDa
HTTP/1.0 Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-electric.com/ws/
ModbusXmlDa/WriteMultipleRegisters"
<?xml version="1.0" encoding="utf-8" ?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/
envelope/">
   <soap:Body>
     <WriteMultipleRegisters xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
       <UnitID>int</UnitID>
       <Address>int</Address>
       <Value>
         <int>int</int>
         <int>int</int>
       </Value>
     </WriteMultipleRegisters>
 </soap:Body>
</soap:Envelope>
HTTP/1.0 200 OKContent-Type: text/xml; charset=utf-8
Content-Length: length <?xml version="1.0" encoding="utf-8" ?>
 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/
envelope/">
  <soap:Body>
     <WriteMultipleRegistersResponse xmlns="http://
www.schneider-electric.com/ws/ModbusXmlDa/">
     </WriteMultipleRegistersResponse>
  </soap:Body>
 </soap:Envelope>
```

ReadCoils

This is the API used to implement a read multiple coils action.

The request provides three parameters:

- the destination of the read request,
- the starting address,
- the number of values to read (base index is provided by starting address).

In response, the list of values is provided.

This API is mapped on Modbus Function Code 01.

Parameters

Input: int UnitID, Address, Quantity

Output: array of int Result

Example

The following is an example of a SOAP request and its response:

```
POST /ws/ModbusXmlDa
HTTP/1.0 Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-electric.com/ws/
ModbusXmlDa/ReadCoils"
<?xml version="1.0" encoding="utf-8" ?>
 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/
envelope/">
   <soap:Body>
     <ReadCoils xmlns="http://www.schneider-electric.com/ws/
ModbusXmlDa/">
       <UnitID>int</UnitID>
       <Address>int</Address>
       <Quantity>int</Quantity>
     </ReadCoils>
   </soap:Body>
</soap:Envelope>

HTTP/1.0 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8" ?>
 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/
envelope/">
  <soap:Body>
     <ReadCoilsResponse xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
       <ReadCoilsResult>
         <int>int</int>
         <int>int</int>
       </ReadCoilsResult>
     </ReadCoilsResponse>
  </soap:Body>
</soap:Envelope>
```

WriteMultiple-Coils	This is the API used to implement a write multiple coils action. The request provides three parameters: ● the destination of the write request, ● the starting address, ● the values to write. This API is mapped on Modbus Function Code 15.
Parameters	Input: int UnitID, Address, array of int Value Output: none

Example

The following is an example of a SOAP request and its response:

```
POST /ws/ModbusXmlDa HTTP/1.0
Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-electric.com/ws/
ModbusXmlDa/WriteMultipleCoils"
<?xml version="1.0" encoding="utf-8" ?>
 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/
envelope/">
   <soap:Body>
     <WriteMultipleCoils xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
       <UnitID>int</UnitID>
       <Address>int</Address>
       <Value>
         <int>int</int>
         <int>int</int>
       </Value>
     </WriteMultipleCoils>
   </soap:Body>
 </soap:Envelope>

HTTP/1.0 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8" ?>
 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/
envelope/">
   <soap:Body>
     <WriteMultipleCoilsResponse xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
     </WriteMultipleCoilsResponse>
   </soap:Body>
</soap:Envelope>
```

ReadInt32 SOAP request/ response

The request provides three parameters:

- The destination of the read request,
- the starting address,
- the number of values to read (base index is provided by starting address).

Parameters

Input: int UnitID, Address, Quantity

Output: array of int Result

Example

In response, the list of values is provided. The following is an example of a SOAP request and response:

```
POST /ws/ModbusXmlDa
HTTP/1.0 Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-electric.com/ws/
ModbusXmlDa/ReadMultipleRegisters"
<?xml version="1.0" encoding="utf-8" ?>
 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/
envelope/">
   <soap:Body>
     <ReadInt32 xmlns="http://www.schneider-electric.com/ws/
ModbusXmlDa/">
       <UnitID>int</UnitID>
       <Address>int</Address>
       <Quantity>int</Quantity>
     </ ReadInt32 >
   </soap:Body>
 </soap:Envelope>

HTTP/1.0 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8" ?>
 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/
envelope/">
   <soap:Body>
     < ReadInt32Response xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
       < ReadInt32Result>
         <int>int</int>
         <int>int</int>
       </ ReadInt32Result>
     </ ReadInt32Response>
   </soap:Body>
 </soap:Envelope>
```

WriteInt32 SOAP request/ response

The request provides three parameters:

- The destination of the write request,
- the starting address,
- the values to write.

Parameters

Input: int UnitID, Address, array of int Value

Output: none

Example

The following is an example of a SOAP request and response:

```
POST /ws/ModbusXmlDa
HTTP/1.0 Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-electric.com/ws/
ModbusXmlDa/WriteMultipleRegisters"
<?xml version="1.0" encoding="utf-8" ?>
 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/
envelope/">
   <soap:Body>
     <WriteInt32 xmlns="http://www.schneider-electric.com/ws/
ModbusXmlDa/">
       <UnitID>int</UnitID>
       <Address>int</Address>
       <Value>
         <int>int</int>
         <int>int</int>
       </Value>
     </ WriteInt32 >
   </soap:Body> </soap:Envelope>

HTTP/1.0 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8" ?>
 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/
envelope/">
   <soap:Body>
   < WriteInt32Response xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
      </ WriteInt32Response>
   </soap:Body>
 </soap:Envelope>
```

SymbolicXMLDA SOAP Services

Overview

The SymbolicXMLDA is the Web Service provided to implement a Symbolic Data Access based on FactoryCast namespace. The services are:

- read,
- write,
- browse.

Limitation on symbolic access

As symbolic access relies on low-level Modbus access for actual communication, only symbols mapped on data accessible through native Modbus requests (Read/Write registers and Read/Write Coils) are supported. Consequently, only data of these types are supported:

- bit
- 16-bit
- 32-bit double word

Unlocated data of all types are not supported.

Read service

The aim of this request is to read one or several symbols. The symbols passed as request arguments are translated using the namespace to get the address of the value to read. When all addresses are known, a MODBUS request is issued by contiguous addresses (i.e., if registers 10, 11, and 13 are requested, 2 requests are issued: one for 10 and 11, and one for 13) and by type of variable desired (coils, registers). Once all the values are retrieved, the SOAP response is built using the variable name, their type and values.

Parameters

Input: list of string ItemName

Output: array of Item ReadResult

Sample

The following is a sample SOAP request and response:

```
POST /ws/SymbolicXmlDa.asmx  HTTP/1
Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-electric.com/ws/
SymbolicXmlDa/1.0/Read"
  <?xml version="1.0" encoding="utf-8"?>
  <soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
    <soap:Body>
      <Read xmlns="http://www.schneider-electric.com/ws/
SymbolicXmlDa/1.0/">
        <ItemList>
          <Items>
            <ItemName>string</ItemName>
          </Items>
          <Items>
            <ItemName>string</ItemName>
          </Items>
        </ItemList>
      </Read>
    </soap:Body>
  </soap:Envelope>
```

Write service

The aim of this request is to write one or several symbols. The symbols, their types, and values are passed as request arguments. The names of the variables are translated using the namespace to get the address of the value to read. When all addresses are known, a MODBUS request is issued by contiguous addresses (i.e., if register 10, 11, and 13 are requested, 2 requests are issued: one for 10 and 11, and one for 13) and by type of variable desired (coils, registers).

Once all the values are written, the SOAP response is built.

Parameters

Input: list of (string ItemName, Value, Type)

Output: none

Sample

The following is a sample SOAP request and response:

```
POST /Recipe/ws/SymbolicXmlDa.asmx HTTP/1.1
Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-electric.com/ws/
SymbolicXmlDa/1.0/Write"
<?xml version="1.0" encoding="utf-8"?>
 <soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
 <soap:Body>
     <Write xmlns="http://www.schneider-electric.com/ws/
SymbolicXmlDa/1.0/">
       <ItemList>
         <Items>
           <ItemName>string</ItemName>
           <Value>string</Value>
           <Type>string</Type>
         </Items>
         <Items>
           <ItemName>string</ItemName>
           <Value>string</Value>
           <Type>string</Type>
         </Items>
      </ItemList>
     </Write>
   </soap:Body>
 </soap:Envelope>
```

Browse service

The aim of this request is to read a list of symbolic variables present on the namespace of the FactoryCast module.

This functionality does not interface with the MODBUS layer since its only action is to return all the namespace items. The response contains all the symbols and their types.

Parameters

Input: list of (string ItemName, Type)

Output: list of Descriptions BrowseResult

Sample The following is a sample SOAP request and response:

```
POST /Recipe/ws/SymbolicXmlDa.asmx
HTTP/1.1 Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-electric.com/ws/
SymbolicXmlDa/1.0/Browse"
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
   <soap:Body>
     <Browse xmlns="http://www.schneider-electric.com/ws/
SymbolicXmlDa/1.0/" />
   </soap:Body> </soap:Envelope>

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length <?xml version="1.0" encoding="utf-8"?>
 <soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
   <soap:Body>
    <BrowseResponse xmlns="http://www.schneider-electric.com/
ws/SymbolicXmlDa/1.0/">
       <Elements>
        <BrowseElement>
           <ItemName>string</ItemName>
           <Type>string</Type>
         </BrowseElement>
         <BrowseElement>
          <ItemName>string</ItemName>
           <Type>string</Type>
         </BrowseElement>
      </Elements>
     </BrowseResponse>
   </soap:Body>
 </soap:Envelope>
```

How to optimize massive symbolic requests

Overview

The symbolic read/write request allows you to perform requests of up to 128 symbols.

You can experience a lag when using a large amount of symbols. There are several rules you can apply to improve your application's speed by reducing the number of internal requests.

The purpose of this topic is to help you create a coherent namespace and use symbolic requests efficiently.

Symbolic request process

Symbolic requests enable the user to use symbols instead of addresses to handle PLC data. This request system uses the namespace on the module to resolve the addresses. Once the addresses are known, there is an optimization process to sort the address by type (coils, registers) in ascending order.

The sorted addresses should be grouped by coherent block to minimize the number of requests sent to the CPU.

The graphic below shows the three steps of the symbolic request optimization:

Optimization method

When you want to perform a massive symbolic request and be efficient in your process, you have to pay close attention to the topology of the symbols:

Step	Action
1	Group the symbols by type: don't mix the coils and registers
2	If you can, sort your symbols in ascending order
3	Try to determine blocks of symbols (i.e., contiguous symbol addresses). These blocks will be done in only one low-level request.

Optimization example

Considering a relatively simple namespace file:

```
|STATION:0.0
16
|Bit_mem1|%M0|0|0|1|1|1|false
|Bit_mem2|%M1|0|0|1|1|1|false
|Bit_mem3|%M2|0|0|1|1|1|false
|Bit_mem4|%M4|0|0|1|1|1|false
|Bit_mem5|%M5|0|0|1|1|1|false
|Bit_mem6|%M6|0|0|1|1|1|false
|Engine_Brake_lvl|%MW0|0|0|4|1|1|false
|Engine_Brake_dsk|%MW1|0|0|4|1|1|false
|Engine_gauge_1|%MW2|0|0|4|1|1|false
|Engine_gauge_2|%MW3|0|0|4|1|1|false
|Engine_gauge_3|%MW4|0|0|4|1|1|false
|Engine_gauge_4|%MW5|0|0|4|1|1|false
|Engine2_Brake_lvl|%MW100|0|0|4|1|1|false
|Engine2_Brake_dsk|%MW101|0|0|4|1|1|false
|Engine2_gauge_1|%MW102|0|0|4|1|1|false
|Engine2_gauge_2|%MW103|0|0|4|1|1|false
|Engine2_gauge_3|%MW104|0|0|4|1|1|false
|Engine2_gauge_4|%MW105|0|0|4|1|1|false
0
1
false
```

There are values for two engines (engine and engine2).

There are also boolean values (Bit_mem).

If the end user asks for every symbol in their namespace order, he will issue one SOAP request with 16 values. As type and addresses (in an ascending order) sort these values, the optimization will only determine blocks of addresses. Here we can see three blocks:

- coils from address 0 to 6,
- registers from address 0 to 5,
- registers from address 100 to 105.

There will be three low-level requests sent to the CPU.

If the end user asks for five symbols in one SOAP request: Engine_gauge_3, Engine_gauge_1, Bit_Mem1, Engine2_gauge_3, Engine2_gauge_1.

Step	Action
1	Since symbols are mixed, the first step of optimization is to sort between coils and registers.

Step	Action
2	In this step, symbols of the same type are sorted by address. As the SOAP request is not well ordered, this step has to process the data. At the end of this step we have two arrays: an array of one coil and an array of four registers:
3	This step should find contiguous addresses. Since there are no contiguous addresses, we have five blocks of one address. The final result is that we have to perform five low-level requests for the CPU.

In brief

When you have to perform intensive use (in number, in time efficiency) of symbolic requests, you must follow these rules:

- group the symbols by types,
- sort the symbols in ascendant order,
- group the symbols by blocks of contiguous addresses.

Visual Basic examples

Overview

To help you start writing your application, this topic shows a Visual Basic example of accessing SOAP requests.

Example

The following example reads ten consecutive registers from register 5.

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim ws As New WindowsApplication26.WebReference.ModbusXmlDa
    ListBox1.DataSource = ws.ReadMultipleRegisters(0, 5, 10)
End SubEnd Class
```

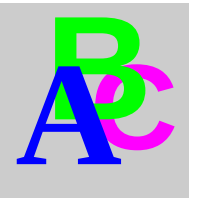
Online SOAP Documentation

Overview

When you open a FactoryCast Web server page in a browser, you can find a **Documentation** hyperlink.

This link takes you to SOAP online documentation, which describes the SOAP Web services available, explains the syntax of SOAP APIs, and provides Visual Basic example.

Index



Numerics

140 NOE 211 10, 15
140 NOE 251 10 (Fiber Optic), 15
140 NOE 771 10, 15
502 (port)
  connections, 44

A

addresses
  deleting, 169
  direct, 169
  editing, 169
  IP, 153
Alarm Viewer display, 338
alarms
  operation and management of, 341
analog I/O diagnostics, 100
analog I/O status
  Micro, 135
applets
  Java, 281
  using, 281
AS-i diagnostics, 102

B

bandwidth
  bandwidth monitoring page, 86
bandwidth monitoring
  diagnostics page, 45
benchmarks, 349
browsers
  requirements, 345
  security, 345, 348
  setting, 345
  versions, 345

C

communication status
  Micro, 133
company logos, 154
configuration tool
  overview, 18
  system requirements, 20
configuration, new, 147
Configure SNMP
  Fields, 65
  Via web server, 64
controller status
  Micro, 132
  Premium, 82
  Quantum, 40
counter status
  Micro, 136
Crash log
  diagnostics page, 49
custom Web pages
  protected, 154, 275
  unprotected, 275
  using Java applets, 281

D

Data Editor
spreadsheets, 196
symbols, 206
templates, 203
variables, 206
data templates, 203
default Web sites
read-only, 27
setup, 147
defaults
restore, 21
Diagnostic pages
Ethernet module, 42
diagnostics
Analog I/O, 100, 140
AS-i, 102
Digital I/O, 98, 139
distributed I/O, 51
FIP I/O module, 92
home page, 80
option modules, 91
PCMCIA communication, 104
remote I/O, 51
standard module, 106
direct addresses, 207
R(ead)/W(rite), 163
read-only, 163
write-enabled, 158, 163, 208
drops page
distributed I/O, 58
dynamic data
controller status, 40
definition, 40
LEDs, 87

E

electronic mail
diagnostics page, 48, 90
electronic mail notification
configuration, 114
service configuration, 114, 115
electronic mail notification service
configuration parameters, 115
parameters, 90
Embedded server
benchmarks
Micro, 349
embedded server
dynamic data, 87
module status, 87
embedded server module
load distribution, 45
status, 45
Embedded Servers, 17
backing up files, 178
benchmarks
Premium, 349
Quantum, 349
checking status, 181
initializing, 179
reflashing, 179
restoring files, 179
Ethernet module
statistics page, 83

F

FactoryCast
diagnostics page, 80
ETY home page, 76
FIP I/O module diagnostics, 92
firewalls, 30
FTP, 17

G

global data
diagnostics page, 42
publish/subscribe, 71
utility, 71

H

home pages
custom, 273

HTTP, 17
  Address server, 109
  Graphic Objects, 317

I

I/O Scanner
  diagnostics page, 43
IP addresses, 153

J

Java applets, 281

L

LED
  dynamic data, 87

M

messaging
  diagnostics page, 44

N

namespaces, 158, 162, 163
  saving, 176
  synchronizing, 170
NOE
  Crash log diagnostics via Web server, 49
  properties via Web server, 50
NOE 771 11
  configuration page, 62
  diagnostics page, 38
  setup page, 62
NTP
  diagnostics page, 46

P

parameters
  setting daylight savings time, 117
  setting NTP server IP address, 117
  setting synchronization period, 117
  setting time zone, 117
passwords, 28, 30, 34, 124, 150
PCMCIA communication diagnostics, 104
performance benchmarks, 349
port 502
  TCP connections, 44
Premium
  FactoryCast module, 76
  home page, 76

Q

Quantum
  controller status, 40

R

Rack Viewer, 137
register values
  valid for Micro, 163
  valid for Premium, 163
  valid for Quantum, 163
restore defaults, 21

S

security, 162, 348
  conflicts, 167
  saving files, 177
SOAP, 369
standard module diagnostics, 106

symbols
accessing, 162
definition, 14
importing, 159
inserting, 206
modifying, 150, 192, 208
read only, 155
setting, 155, 162
setting access, 162
viewing, 192
write enabled, 155
system requirements, 20

T

TCP/IP
connections (open/closed), 44
time synchronization service
parameters, 88
time zones, 119
customizing, 118
traps
setting, 117
TSX ETY 110 WS, 15
TSX ETY 510
diagnostics page, 80
home page, 76
TSX ETZ 510, 16
tz compilers, 118

V

variables
accessing, 162
definition, 14
importing, 159
inserting, 206
modifying, 150, 192, 208
read only, 155
setting, 155, 162
setting access, 162
viewing, 192
write enabled, 155

W

Web pages
NTP Configuration, 116
Web server
Configure SNMP, 64
Crash log diagnostics via Web server, 49
NOE Properties Page, 50